Naval Research Laboratory
Stennis Space Center, MS 39529-5004

NRL/MR/7241--95-7577

Documentation for the Semi-Automated Mesoscale Analysis System 1.2

SARAH H. PECKINPAUGH

Remote Sensing Applications Branch
Remote Sensing Division

April 19, 1995

REPORT DOCUMENTATION PAGE

Form Approved
OBM No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 19, 1995	Final

4. TITLE AND SUBTITLE

Dcumentation for the Semi-Automated Mesoscale Analysis System 1.2

5. FUNDING NUMBERS

Job Order No. 572519705
Program Element No. 0603207N
Project No. X1596
Task No.
Accession No. DN394464

6. AUTHOR(S)

Sarah H. Peckinpaugh

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
Remote Sensing Division
Stennis Space Center, MS 39529-5004

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/7241--95-7577

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Space and Naval Warfare Systems Command
2451 Crystal Drive
Arlington, VA 22245-5200

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Several software modules have been developed for or by the Naval Research Laboratory to automate the analysis and interpretation of satellite infrared imagery in the Gulf Stream region of the northwest Atlantic Ocean. The modules have been integrated to form the Semi-Automated Mesoscale Analysis System version 1.2 (SAMAS). This document provides the information required for an analysis to use SAMAS 1.2. This document provides descriptions of what the modules of SAMAS 1.2 do and describes user inputs and outputs. This document also provides information on the location of common source codes required for program development and data formats for input/output files.

14. SUBJECT TERMS

satellite remote sensing, expert system, artificial intelligence, automated analysis

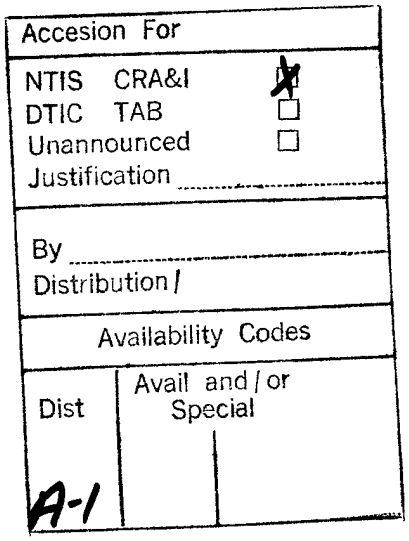
15. NUMBER OF PAGES

92

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

Table of Contents



DOCUMENTATION FOR THE SEMI-AUTOMATED MESOSCALE ANALYSIS SYSTEM 1.2

1.0 Introduction

Several software modules have been developed for or by the Naval Research Laboratory (NRL) to automate the analysis and interpretation of satellite infrared (IR) imagery in the Gulf Stream region of the Northwest Atlantic Ocean. These modules have been integrated to form the Semi-Automated Mesoscale Analysis System version 1.2 (SAMAS), see Fig. 1. SAMAS starts with a satellite IR image, which can be a warmest pixel composite with an optional cloud mask. From these an edge image is created. The edges are labeled by one of two methods, relaxation or topological. Each labeling method requires for input the original input image and a previous Gulf Stream and eddy analysis that has optionally been progressed in time by the Oceanographic Expert System. An edge/region segmentation process can be applied for use as an analyst tool. Using a linear hough transform, Gulf Stream altimeter crossings can be defined. These crossings can be input to the edge labeling modules and into the next step, the Gulf Stream formation process. Given the labeled edges and any altimeter Gulf Stream crossing that has been defined, SAMAS has modules to interpolate the Gulf Stream using complex empirical orthogonal functions (CEOF) and to define eddies using a circular hough transform. The Expert System can be used to progress eddies and Gulf Stream positions from the previous analysis. This is very useful for filling areas where cloud cover prevents a clear view of the features at the current time. The Gulf Stream, the eddies, and other frontal edges can be viewed and edited by an analyst using an interactive editor designed especially for SAMAS. The final output of the system is a chart showing the locations of mesoscale features for the input image.

2.0 SAMAS Driver and Functions

SAMAS has been implemented to run on a SUN SPARC work station. The system can be split into two categories: interactive editor and noneditor functions. The interactive editor has been implemented as part of the NRL Satellite Image Processing System (NSIPS) running under the Precision Visuals-Workstation Analysis and Visualization Environment (PV-WAVE) command language. For the noneditor modules a menu driver has been developed. This menu driver is implemented within the PV-WAVE environment. The noneditor functions can be run as independent executables when desired. With the exception of the Expert System, the independent modules do not require PV-WAVE to run. The Expert System allows the user an interactive display capability which if chosen uses PV-WAVE. The Expert System needs to be run separate from the menu if the display option is desired; this is a limitation of the system.

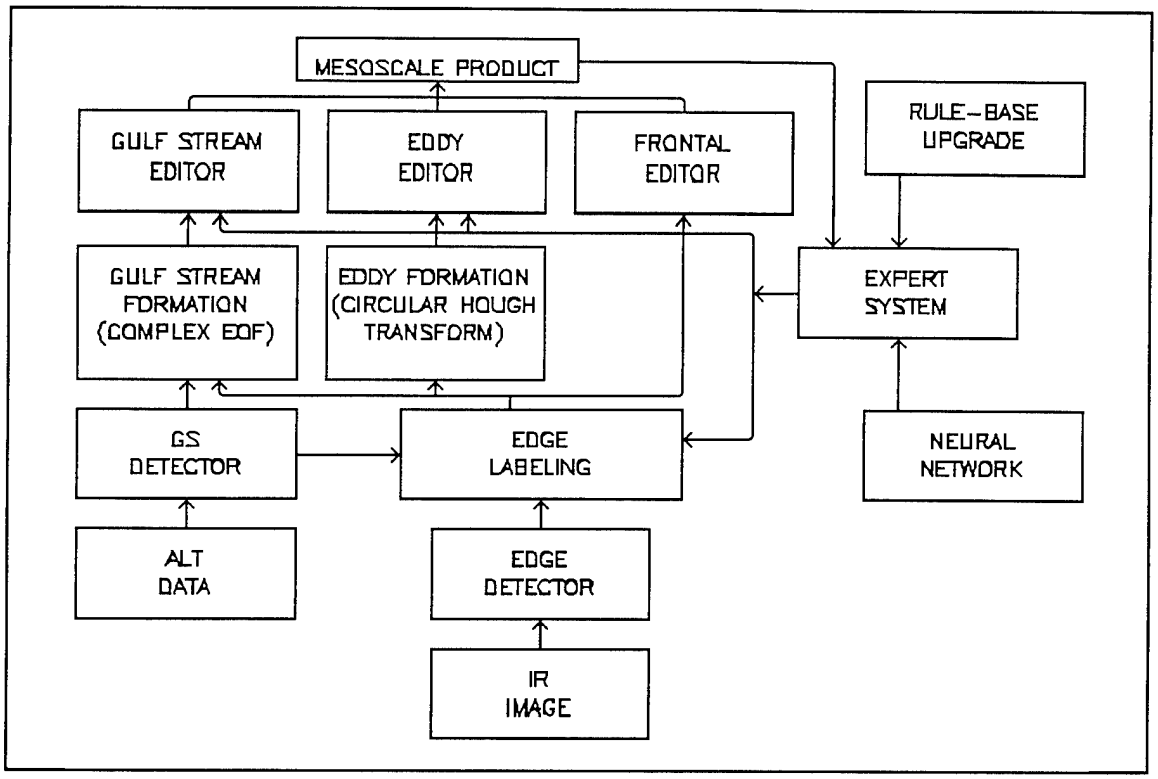


Figure 1. A functional diagram of SAMAS 1.2.

2.1 SAMAS Editor

The editor is part of the NSIPS. For a complete descriptions of NSIPS see the NSIPS User's Guide [10]. The main menu for NSIPS has the option "User Functions". It is this option that will provide the user with the menu containing the SAMAS option. Selecting the SAMAS option activates the interactive editor. The editor can be divided into 3 main functions with other auxiliary function available. The three main functions are Gulf Stream editor, frontal editor, and eddy editor. For a more complete description of these see Appendix A.

2.2 SAMAS Noneditor Functions

The remaining functions of SAMAS can be accessed as independent modules or via a PV-WAVE function which supplies the user with an easy-to-use menu. To define the environment variables for SAMAS processing you must source the .defines file (see App. C). To access the menu start PV-WAVE in the usual manner (type "wave"), then compile and use samas.pro. The commands are:

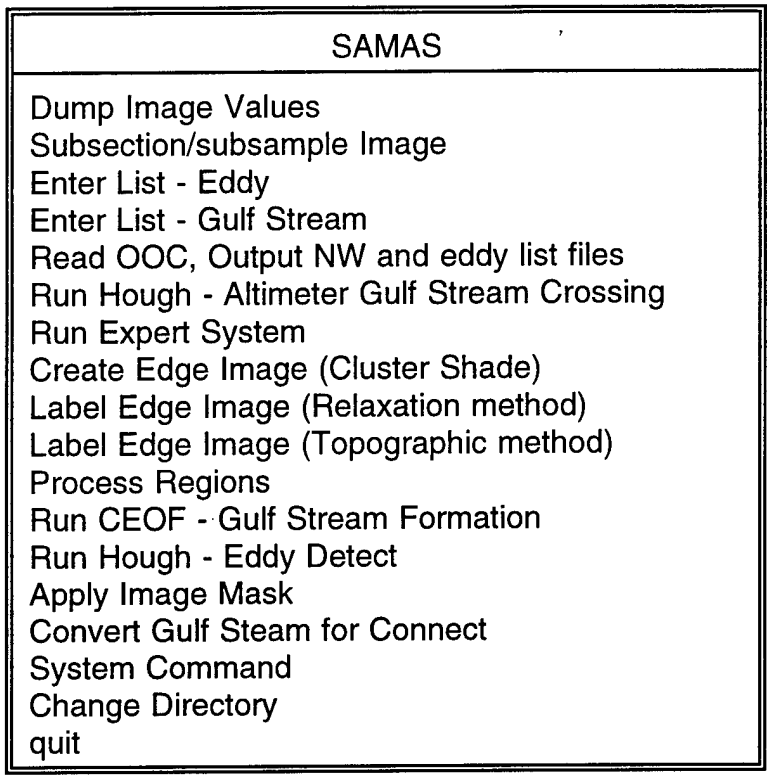


Figure 2. SAMAS menu.

```
:source /disks/sips3d2/sarah/.defines

wave> .rnew $SRCU/samas.pro
wave> .rnew $SRCU/samas.pro
wave> samas
```

See Fig. 2 for the SAMAS menu. The user can then move the mouse to the desired option and click. Each option contained in the menu is described below, along with the file name of the executable required to run the module independent of the menu system.

2.2.1 Dump Image Values

The user can select this option from the menu, or the command $SAMASEXE/dump_image can be used. This function is used to dump the header information and data values from SAMAS image files. The user provides an input file

name. The program prints the header information: number of samples, number of lines, data type, north latitude limit, west longitude limit, degree/pixel ratio, and type of map projection. The user will then be asked for "starting sample and line - ". The user must enter two values separated by a blank. Starting at the specified sample and line, ten lines of ten samples will be printed for the user. The user will be reprompted for "starting sample and line - " and the desired data printed, until zero values are entered at the prompt. The program prompts are:

Input image name -

The user enters the name of the input SAMAS image file.

starting sample and line -

The user enters the sample and line coordinates of the upper-left corner of the 10 x 10 pixel box of data to be dumped. The user will be reprompted for these values until values of zero are input to end the program run. The values should be separated by a blank, not a comma.

An example run of this program follows:

```
Input image name -  w_test.img
Nsamp - 70
Nline - 70
Dtype - Byte
North - 45.000000
West - -75.000000
DegPixel - 0.025000
Proj - Merc

starting sample and line - 1 1

Rec 1 samp 1
      183   175   180   182   169   122   182   192   183   173
      171   186   174   125   168   179   179   186   171   170
      145   183   162   159   147   182   187   175   179   169
      184   188   171   167   168   179   183   177   149   160
      188   192   174   179   183   186   186   170   155   170
      187   158   175   184   196   186   177   173   170   172
      181   184   184   187   183   177   177   162   161   169
      186   184   183   182   175   181   162   172   135   210
      190   182   183   185   178   191   162   173   180   124


starting sample and line - 0 0
```

2.2.2 Subsection/subsample Image

The user can select this option from the menu or the command $SAMASEXE/ss_image can be used. This function allows the user a way of reducing the size of an image. The image can be limited by area and/or subsampled. The user controls the process by specifying starting sample, starting line, number of samples, number of lines, and sampling factor. The output image can be either a SAMAS image file with the header corrected for the new mapping information, or the output can be a row/column type output file. The program prompts are:

Input image name -
The user is prompted for the name of the input SAMAS image file.

Desired starting sample (def: 1) -
The user supplies the one relative starting sample number for extracting the image area.

Desired number of output samples (def: ??) -
The user supplies the desired number of output image samples, after subsectioning. The default is computed given the number of input samples and the desired starting sample, assuming a sampling factor of one.

Desired starting line (def: 1) -
The user supplies the one relative starting line number for extracting the image area.

Desired number of output lines (def: ??) -
The user supplies the desired number of output image lines, after subsectioning. The default is computed given the number of input lines and the desired starting line, assuming a sampling factor of one.

Desired sampling increment (def: 1) -
The user supplies the subsampling increment to be used.

Desired output data type (1-byte, 2-short, 3-int, 4-real) (def: ??) -
The user selects the desired output data type. The default will be the input data type.

Output image name -
The user supplies the output image file name.

Output image type (def: img or rc) -
The user selects the desired type of output image file, SAMAS or row/column.

2.2.3 Enter List - Eddy

The user can select this option from the menu or the command $SAMASEXE/eddy_enterlist can be used This function allows the user to create an eddy list file. The user can optionally input an eddy list file. The eddies from the input file will be written to the new output eddy list. The user can choose to enter the eddy radius values in either kilometers or nautical miles. The user is prompted for a two character ASCII source code. This code is strictly for the user's reference. Some eddy-creating modules of SAMAS output source codes unique for the function, example: ED for Eddy Detect. The user is prompted for latitude, longitude, radius, and eddy type (Warm, Cold, or Undefined) until no value is entered for the prompt. Each eddy will have the previously entered source code. The user is prompted again for a source code. If no value is entered, then the new eddy list file is created from the data which has been entered by the user. If a new source code is entered, then the user is again prompted for more eddy information until none is entered. This process continues until no value is entered for source code. As eddies are entered into the list, they are checked against the eddies already contained in the list. If a new eddy is found to overlap any of the eddies already in the list, then the new eddy will be kept and the other eddy or eddies will be deleted from the list. No overlapping of eddies is allowed. The program prompts are:

Input File -

The user is prompted for an optional input eddy list file. If a file is supplied, then the eddies of this file are written to the output list and the new eddies to be entered by the user will be appended to the list. If a new eddy overlaps an existing eddy, then the existing eddy will be deleted from the list.

Output File -

The user is prompted for an output eddy list file. This file will contain the newly created eddy list.

Radius values will be KM (def: K) or Nautical Miles (N) -

The user is prompted for the unit of measure for the eddy radius values to be entered. If the values are in nautical miles, then they will be converted to kilometers for output.

Enter Source Code -

The user is prompted for a two character ASCII source code. This code is used to define the origin of the eddy information. This code is used for all eddies entered until a new code is entered. If the user enters no code, then the output file is written and the program ends.

Enter Lat, Lon, Rad, Type -

The user enters these values along with the source code to define an eddy. The user enters the latitude, longitude, radius (in the units specified km or nautical miles), and type (Warm, Cold, or Undefined). If the user enters no values at this prompt, then the prompt for source code is given.

An example run of this program follows:

```
Input File -
Output File - Apr1.eddy
Radius values will be KM (def: K) or Nautical Miles (N) -
Enter Source Code -  OC
Enter Lat, Lon, Rad, Type - 39.3, -70.3, 40, W
Enter Lat, Lon, Rad, Type -  40.0, -65.6, 39, W
Enter Lat, Lon, Rad, Type -
Enter Source Code -  GP
Enter Lat, Lon, Rad, Type - 35.2, -67.2, 45, C
Enter Lat, Lon, Rad, Type -
Enter Source Code -
```

	Latitude	Longitude	Radius	Type	Source
1	39.300	-70.300	40.000	W	OC
2	40.000	-65.600	39.000	W	OC
3	35.200	-67.200	45.000	C	GP

2.2.4 Enter List - Gulf Stream

The user can select this option from the menu or the command $SAMASEXE/gs_enterlist can be used. This function allows the user to create a Gulf Stream list file. The user can optionally input a Gulf Stream list file. The positions from this file will be written to the new Gulf Stream list. The user is prompted for a single character ASCII source code. This source code defines the origin of the data. The codes used are "I" for image, "A" for altimeter, and "O" for other. Points with a code of I are expected to be points of a continuous stream instead of isolated positions. The user is prompted for latitude and longitude until no value is entered for the prompt. Each position will have the previously entered source code. The user is prompted for a new source code. If no value is entered, then the new Gulf Stream list file is created. If a new source code is entered, then the user is again prompted for more Gulf Stream positions until no value is entered. This process continues until no value is entered for source code. The program prompts are:

Input File -

The user is prompted for an optional input Gulf Stream list file. If a file is supplied, then the positions from this file are written to the output list and the new positions entered by the user will be appended to the list.

Output File -

The user is prompted for an output Gulf Stream list file. This file will contain the newly created Gulf Stream list.

Enter Source Code -

The user is prompted for a single-character ASCII source code. This code is used to define the origin of the position information, and is used for all positions entered until a new code is entered. If the user enters no code, then the output file is written and the program ends.

Enter Lat, Lon -

The user enters the next latitude, longitude position of the Gulf Stream. If the user enters no values at this prompt, then the prompt for source code is given.

An example run of this program follows:

```
Input File - Apr1.NW.gs
Output File - Apr1_2.NW.gs
Enter Source Code - I
Enter Lat, Lon - 26.8, -79.9
Enter Lat, Lon - 27.3, -80
Enter Lat, Lon - 28.4, -80.4
Enter Lat, Lon - 29.4, -80.5
Enter Source Code -
```

2.2.5 Read OOC, Output NW and eddy list files

The user can select this option from the menu or the command $SAMASEXE/reformat_ooc can be used. This function is used to reformat mesoscale analysis received from the Warfighting Support Center (WSC) of the Naval Oceanographic Office (standard front and eddy file format version 5.01). The Gulf Stream North Wall, Gulf Stream South Wall, and eddy coordinate information that meets the user-specified confidence threshold is extracted from the input file. The confidence threshold allows the user to distinguish between estimated positions and observed positions of the mesoscale features. This function outputs three files: Gulf Stream North Wall list file, Gulf Stream South Wall list file, and eddy list file. The output file names are created

automatically from the input WSC file name: *.nw.gs, *.sw.gs, and *.eddy, where * is the WSC file name.

Input OOC file -

The user enters the WSC analysis file name to be read (note: WSC formerly called OOC).

Input confidence threshold -

The user enters the desired confidence threshold to be used for selection of positions to be written to the output files. The range of values is 0 - 100.

2.2.6 Run Hough - Altimeter Gulf Stream Crossing

The user can select this option from the menu or the command $SAMASEXE/houghgs can be used. This function finds the North Wall Gulf Stream crossing for a given altimeter track. The computed crossing can be output to a Gulf Stream list file or the output can be appended to an existing Gulf Stream list file. The method uses a linear hough transform for this process. The program prompts are:

Input Gulf Stream file (opt.) -

This is an optional input. If an input Gulf Stream list file is provided, then the new Gulf Stream crossing points will be appended to the input list and the entire list will be output.

Output Gulf Stream file -

The user must provide a name for the output Gulf Stream list file.

Enter track file name -

The user must supply an input track. To this will be appended the suffix ".dat" to form the file name. This file contains a list of longitude, latitude positions for the track.

Enter orbit file name -

The user must supply an input orbit. To this will be appended the suffix ".dat" to form the file name. This file contains a list of latitude, SSH residuals for the orbit.

2.2.7 Expert System

The user can select this option from the menu or the command $EXPERTEXE/expsys can be used. The user is given options for parameter prompting: noninteractive, interactive, or options-window-interactive. The interactive modes allow the user an optional PV-WAVE graphic display and options for the output of rule explanations for eddy

motion. This module should never be run via the SAMAS menu program if PV-WAVE output is desired (a conflict will occur). The executable name should be used when PV-WAVE output is desired. This function accepts as input: a Gulf Stream North Wall (opt.) and an eddy list file. If the North Wall is not provided, then the default nominal North Wall will be used. The South Wall is generated to be 100 km south of the North Wall. The user supplies a value for number of days for incrementing the forecast. The North Wall and eddies will be progressed in time by the number of days specified. In the interactive mode the user can step, incrementing by the number of days, until quit is selected. The user has the option of keeping any or all three of the following predicted outputs: North Wall, South Wall, or eddies. The Gulf Stream predictions are computed using executable code generated by a trained backpropagation neural network [9]. The eddy predictions are computed via a rule system. Gulf Stream positions output by this function will have a source code of "O". Eddies output by this function will have a source code of "ES". The PV-WAVE output allows the user some choices for background and what is to be plotted. For a more complete description of the explanation capabilities of the system see Ref. 1. For more information on the Expert System in general, see Ref. 3.

The initial program prompt is:

Prompting (def: Non-interactive, Interactive, or Options-Window-Interactive) -

The user has the option of Non-interactive, Interactive, or Options-Window-Interactive. Non-interactive means the user will be prompted for inputs, the program runs, and the output file(s) are created. For Interactive mode, the user will be prompted as needed for stepping the day increment. Also, explanations for eddy movement can be printed. The Options-Window-Interactive supplies the user with a window for inputs to the program. The program can be started again and again from the window, where parameters can be easily changed for each new run. The user is prompted for stepping and eddy movement explanations.

For Non-interactive mode, the prompts to the user are:

Input North Wall Gulf Stream -

The user is prompted for the North Wall Gulf Stream file (upper) or the nominal North Wall can be used if no file is specified. This input file should be in the SAMAS format for Gulf Stream files. The header record contains the number of positions. The remaining records contain the latitude, longitude, and source code for the Gulf Stream positions. This program does not use the source code.

Input Eddy -

The user is prompted for the input eddy list file. This input is optional. This file should be in the SAMAS format for eddy files. The header record contains the number of eddies. The remaining records contain the latitude, longitude, radius,

type, and source code for the eddies. This program does not use the source code.

Input day increment -

The user is prompted for the number of days to progress the input Gulf Stream and eddies.

Output North Wall Gulf Stream -

The user can optionally choose to have the new North Wall Gulf Stream output to a SAMAS Gulf Stream list file. This file will be in the SAMAS Gulf Stream file format. The program will output a source code for these new positions of "O".

Output South Wall Gulf Stream -

The user can optionally choose to have the new South Wall Gulf Stream output to a SAMAS Gulf Stream list file. This file will be in the SAMAS Gulf Stream file format. The program will output a source code for these new positions of "O".

Output Eddy -

The user can optionally choose to have the new eddies output to a SAMAS eddy list file. This file will be in the SAMAS eddy file format. The program will output a source code for these new positions of "ES".

See Fig. 2 for program input window for Options-Window-Interactive. The buttons are selected using the mouse. Special buttons unique to this mode are "OK", "QUIT", and "HELP". OK means start the program; multiple runs can be made using this mode. QUIT terminates the processing and dismisses the window, and HELP is self-explanatory. The explanations for the other prompts are the same as those described below for Interactive mode. Prompts for Interactive mode are:

Use PV wave (y or n)?

The user can select an optional PV-WAVE display. Rings and/or Gulf Stream will be plotted on a map background.

Draw rings (y or n)?

If a PV-WAVE graphic output is specified, then the user can choose to have new rings drawn onto the map background.

Draw ring trails (y or n)?

If the user has selected to draw rings, then the user can choose to have the ring trails saved on the map background in a second color for viewing.

Draw GS (y or n)?

If a PV-WAVE graphic output is specified, then the user can choose to have the new Gulf Stream drawn onto the map background.

Erase previous GS (y or n)?

If the user has selected to draw the Gulf Stream, then the user can choose to have the Gulf Stream trail erased or saved on the map background in a second color for viewing.

Draw grids only (y or n)?

Draw regions only (y or n)?

When using the PV-WAVE display, the user can select as part of the map background a latitude-longitude grid, Expert System region boundaries, or neither be shown.

Read in Gulf Stream boundary points from UGS file (y or n)?

name of UGS file =

The user is prompted for the input upper Gulf Stream file (North Wall) or the nominal North Wall can be used. This input file should be in the SAMAS Gulf Stream file format. This program does not use the source code.

name of eddy file =

The user is prompted for the input eddy list file. This file should be in the SAMAS eddy file format. This program does not use the source code.

Updating by day steps of

The user is prompted for the number of days to progress the input Gulf Stream and eddies per cycle.

The following prompts are used for both the Options-Window-Interactive and Interactive modes:

Enter s to step or q to quit

The user selects step another cycle or quit.

******Explanation Menu*****

Eddies are listed here by number with an option for "all eddies" and "quit"

Enter response>

The user is provided a list of eddies, all, and quit. The user can choose to see information for any or all eddies processed. The prompt will be repeated until zero or the proper number for the quit option is chosen.

Rule trace (r) or Summary (s) >

When an eddy or all eddies is selected from the list the user will be prompted for the type of report: rule trace or summary. Output will be to the screen for a single eddy and to a file for all. The output file name will be printed on the screen for the user.

Create output North Wall Gulf Stream file (y or n)?
Final output North wall Gulf Stream file -

The user can choose to have the new North Wall Gulf Stream output to a SAMAS Gulf Stream list file. If yes, the user is prompted for the output file name. This file will be in the SAMAS Gulf Stream file format. The program will output a source code for these new positions of "O".

Create output South Wall Gulf Stream file (y or n)?
Final output South wall Gulf Stream file -

The user can choose to have the new South Wall Gulf Stream output to a SAMAS Gulf Stream list file. If yes, the user is prompted for the output file name. This file will be in the SAMAS Gulf Stream file format. The program will output a source code for these new positions of "O".

Create output eddy list file (y or n)?
Final output Eddy file -

The user can choose to have the new eddies output to a SAMAS eddy list file. If yes, the user is prompted for the output file name. This file should be in the SAMAS eddy file format. The program will output a source code for these new positions of "ES".

2.2.8 Create Edge Image (Cluster Shade)

The user can select this option from the menu or the command $SAMASEXE/edge can be used. The user may choose to run the functions used to create the edge image or to create a shell script that can be used later to create the edge image. The edge function consists of four modules. This driver will provide the prompts required to execute all or some of the modules. Intermediate output files will be created and deleted for the user. The output file of one module will be the input for the next and only the last output image will be saved. The user may elect to run these modules separately and keep or delete the intermediate image files as desired. The user is prompted for which of the modules are to be run. If the user selects to run cluster shade and any other module, then cluster test must also be run. The output from cluster shade can only be input to the function cluster test. The user may optionally clean, dilate, or thin the edge image. The clean and dilate are included in the same module, but either can be done without the other. The input for cluster shade is an IR image with an optional mask, perhaps for clouds or land. The input for cluster test is the output image from cluster shade. The inputs to the remaining modules can be any edge image in the SAMAS format. A prompt is given for a working directory. This is where the working files will be created, and if no error occurs they will be deleted. The user will be prompted for the parameters appropriate for the selected modules (for more information on this type of edge detection see Refs. 4 and 5). The initial prompts for creating an edge image are:

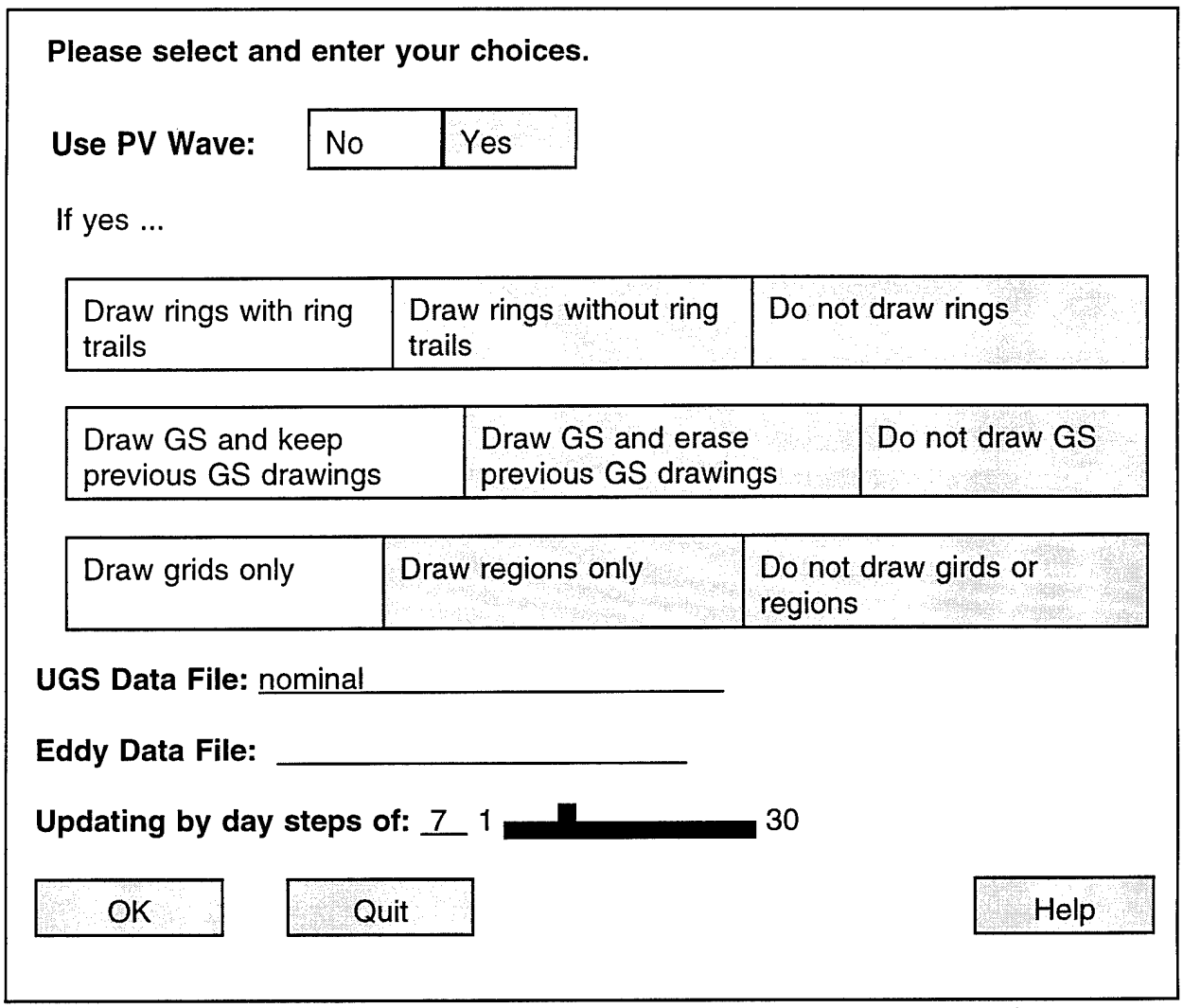


Figure 3. Parameter window for Options-Window-Interactive mode.

Process or create shell script (def: Proc or Com) -

The user is prompted for the type of run. If Proc (process) is selected, then the programs selected by the user will be executed as soon as all the inputs have been entered. If Com (command) is selected, then the user will be prompted for a shell script name and a shell script file will be created. This shell script can be used for batch processing the creation of the edges.

Name for shell script (def: Edge.Com) -

If the user has selected to create a shell script, then this prompt is for the name of the shell script. A default name has been provided.

Run Cluster Shade (def: YES or NO) -

The user is prompted for Yes or No for running the cluster shade module. This module creates an image of cluster shade values.

Run Cluster Test (def: YES or NO) -

The user is prompted for Yes or No for running the cluster test module. This module creates edges from a cluster shade output image.

Run Clean (def: YES or NO) -

The user is prompted for Yes or No for running the clean option of the dilate/clean module. Any edge image can be used as input to this module.

Run Dilate (def: YES or NO) -

The user is prompted for Yes or No for running the dilate option of the dilate/clean module. Any edge image can be used as input to this module.

Run Thin line (def: YES or NO) -

The user is prompted for Yes or No for running the thin line module. Any edge image can be used as input to this module.

Input image file -

The user is prompted for an image file to be used as input for the first module selected.

Output image file -

The user is prompted for an output image. This image will be the output of the last module selected.

Working directory for temporary files (def: current directory) -

The user is prompted for a directory for creating working files that are the output from one of the selected modules which is to be the input to the next selected module.

2.2.8.1 Cluster Shade

This algorithm computes an output image of cluster shade measures. These measures are derived from grey level co-occurrence (GLC) matrices computed on the overlapping windows of the input image. The (i,j)th element of the GLC matrix, $P(i,j \mid \Delta x, \Delta y)$, is the

relative frequency with which two image elements, separated by distance ($\Delta x,\Delta y$), occur in the image window, one with intensity level i and the other with intensity level j. Consider an M x N pixel image window with L intensity levels ranging from 0 to (L-1). Let f(m,n) denote the intensity level of pixel (m,n). Then

$$P(i,j|\,\Delta x,\Delta y) = \sum_{i=1}^{m-\Delta x} \sum_{j=1}^{n-\Delta y} A$$

where A = 1 / (M-Δx X N-Δy) if f(m,n)=i and f(m+Δx,n+Δy) = j. Otherwise, A=0. P(i,j$|\,\Delta x,\Delta y$) is therefore an L x L matrix of second order probabilities. In uniform areas of the image, two pixels displaced by ($\Delta x,\Delta y$) are likely to have nearly equal intensities, i.e., i=j, which means that these pixels contribute to increased probabilities in the near-diagonal elements of the GLC matrix. By contrast, in areas of the image where the displacement ($\Delta x,\Delta y$) spans an edge, i.e. i<<j or i>>j, the probabilities in the off-diagonal elements of the GLC matrix are increased accordingly. The cluster shade measure, S($\Delta x,\Delta y$) is computed for each GLC matrix as follows:

$$S(\Delta x,\Delta y) = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} (i+j-\mu_i-\mu_j)^3 \, P(i,j|\ \Delta x,\Delta y)$$

where,

$$\mu_i = \sum_{i=0}^{L-1} i \sum_{j=0}^{L-1} P(i,j|\ \Delta x,\Delta y)$$

$$\mu_j = \sum_{i=0}^{L1} \sum_{j=0}^{L1} j \, P(i,j|\ \Delta x,\Delta y)$$

The center point of each window of the image is replaced by the S($\Delta x,\Delta y$) value computed from its neighbors, thus creating a "cluster shade" image of S($\Delta x,\Delta y$) values. Points that are on the output image edge, thus not the center of any window, are set to zero.

The user may chose not to compute GLC matrices, or cluster shade measures, for all windows of the input image. There are two methods allowed for eliminating pixels from

the computations. If a mask input image is provided, then the nonzero values of this image will be used to mask pixels. No pixel of the original input image will be processed if there is a corresponding nonzero pixel in the mask input image. The second method is to eliminate pixels in the first input image that do not meet the input processing threshold. If a positive threshold is specified, then no pixels with intensity greater than or equal to the threshold will be processed. If a negative threshold is specified, then no pixels with intensity less than or equal to the absolute value of the threshold will be processed. At least 75% of the pixels in a window must be considered good for a cluster shade value to be computed, else a zero will be output for that window. The prompts for running this option are:

Input mask image for Cluster Shade (opt.) -

The user is prompted for an optional input image to be used for masking. The algorithm does not process neighborhoods with less than 75% nonmasked values.

Window size x dimension (def: 16) -

The user is prompted for the number of samples in the neighborhood or processing window to be used for computing the GLC matrix, which is used to compute the cluster shade measure.

Window size y dimension (def: 16) -

The user is prompted for the number of lines in the neighborhood or processing window to be used for computing the GLC matrix which is used to compute the cluster shade measure.

Delta length x dimension (def: 0) -

The user is prompted for the number of samples in the displacement vector to be used for computing the GLC matrix which is used to compute the cluster shade measure.

Delta length y dimension (def: 0) -

The user is prompted for the number of lines in the displacement vector to be used for computing the GLC matrix which is used to compute the cluster shade measure.

Input processing threshold -

The user is prompted for an optional processing threshold. If the threshold is positive, then the algorithm does not use any values of the neighborhood, that are greater than or equal to this threshold. If the threshold is negative, then the algorithm does not use any values of the neighborhood that are less than or equal to the absolute value of this threshold. The algorithm does not process neighborhoods with less than 75% nonmasked values.

2.2.8.2 Cluster Test

This function creates an edge image based on finding certain zero crossings in the input cluster shade image. The first step is to find the zero crossings that satisfy the threshold specified as the initial threshold. Every pixel of the input cluster shade image is tested. If the absolute value of the pixel is greater than or equal to the initial threshold, then the eight immediate neighbors of that pixel are tested. If any of the eight immediate neighbors also have an absolute value greater than or equal to the initial threshold and are opposite in sign (+,-) from the center pixel of the 3 x 3 neighborhood, then an edge has been found. The center pixel of the 3 x 3 neighborhood is marked with a value of 255 in the output edge image. For no edge a value of zero is output.

If the parameter for maximum passes has not been set to zero, then additional passes will be made through the image to attempt to extend some of the edges which were created by doing the initial zero crossing test. If a pixel has not yet been marked as an edge and has a neighbor that has been marked as an edge in a previous pass, then that pixel is tested for a zero crossing using the minimum threshold. If the pixel passes this zero crossing test, then the pixel is marked as an edge pixel. After each pass, the number of newly added pixels is listed for the user. These passes are repeated until no new pixels are being added or a specified maximum passes have been made through the image to extend the edges. The output image is a byte image with values of zero for the background and 255 for the edges. The prompts for this option are:

Initial threshold (def: 20) -

This value is used to perform the initial zero crossing test of the cluster shade measures. If a pixel has an absolute value greater than or equal to the initial threshold and any of the eight immediate neighbors of that pixel also have an absolute value greater than or equal to the initial threshold and are opposite in sign from the center pixel of the 3 x 3 neighborhood, then a zero crossing has been found.

Minimum threshold (def: 5) -

This value is used to perform zero crossing tests for pixels that have not been marked as an edge but have any of its eight immediate neighbors that have been marked as an edge in any previous pass through the image.

Maximum allowed passes (def: 30) -

After performing the initial zero crossing test on the pixels of the input cluster shade image, the image will be processed again to extend the edges. The process of extending edges will be performed until no new edge pixels can be assigned or the process has been performed the maximum allowed extra passes as specified here.

2.2.8.3 Dilate/Clean

This function performs two tasks, first to clean and then dilate (or thicken) an edge image. The user defines the size of the clean window. The clean window is moved over the entire input image. If a nonzero pixel or group of nonzero pixels is completely contained within the window with no nonzero pixels on the boundary of the window, then the window is cleaned. A window is cleaned by setting all pixels contained within the window to 0.

The second task of this function is dilation. This is done to thicken lines and, as a result, in some cases lines will be connected in the process. The user defines the number of times dilation will be performed on the input image. For each time of dilation all pixels are checked. If a pixel is nonzero, then that pixel and all eight of its immediate neighbors will be set to one. A line 1 pixel wide at the start will be 3 pixels wide after one step, 5 pixels wide after two steps, etc. The prompts for this option are:

Window size x dimension (def: 16) -

Number of samples in the window to be used for image cleaning. The range for this parameter is 1 - 16.

Window size y dimension (def: 16) -

Number of lines in the window to be used for image cleaning. The range for this parameter is 1 - 16.

Number of dilate steps (def: 1) -

Defines the number of iterations of dilation or line thickening to perform.

2.2.8.4 Thin Line

This function, given a binary input image, produces a skeleton of that image. All features of the original image are reduced to single-pixel-wide lines. Outer layers are peeled off, while endpoints and connectivity are preserved. See Ref. 6 for more information on the method used.

Given the following notations and definitions:

notation:		
	Q ...	The set to be thinned
	S ...	The skeleton
	B(Q) ...	The contour of Q
	L(Q) ...	The set of pixels of B(Q) that are not multiple pixels
	M(Q) ...	The set of pixels of B(Q) that are multiple pixels
	K(Q) ...	The set of all pixels in B(Q) that have neighbors in S

4 3 2 D-neighbors are 1,3,5,7
5 P 1 I-neighbors are 2,4,6,8
6 7 8

Definition 1. The contour of a set of pixels Q is defined as the set of pixels in Q which have at least one D-neighbor not in Q.

Definition 2. The C-neighbor of a pixel on a contour are defined as the previous and next pixels (possibly coinciding) found during contour tracing.

Definition 3. A pixel is said to be multiple if one or more of the following conditions hold:

(A) It is traversed more than once during tracing (i.e., at the completion of the process its value is greater than 2 if using actual tracing method).
(B) It has no neighbors in the interior of the region (i.e., no neighbor with a value of 1).
(C) It has at least one D-neighbor, which belongs to the contour but is not one of its C-neighbors.

Definition 4. A skeletal pixel is one for which one of the following conditions is true:

(A) It is a multiple pixel.
(B) It has a D- or I-neighbor which has been identified as a skeletal pixel during an earlier tracing.
(C) (optional) Its two C-neighbors form an angle of 90°.

The steps of the algorithm are:

I. Set S to the empty set
II. while Q is not empty do steps III - VI.
Begin
III. Find B(Q), by contour tracing, use neighborhood check to compute B(Q) for all pixels of Q with a D-neighbor not an element of Q.
IV. Find L(Q) and M(Q) by retracing B(Q) while checking for conditions of Definition 3.
V. Find K(Q) by examining all pixels in B(Q) for neighbors in S and, if desired also checking condition (C) of Definition 4.
VI. Set S = S U M(Q) U K(Q) and Q = Q - B(Q).
END
VII. End of algorithm.

2.2.9 Label Edge Image (Relaxation method)

The user can select this option from the menu or the command $SAMASEXE/label_driver can be used. The user may choose to run the functions used to create the labeled output or to create a shell script that can be executed later to create the labeled output. Feature labeling is done by a method of nonlinear probabilistic relaxation. The user supplies an IR image, an edge image created from the IR image, and a previous analysis for any or all of the following: North Wall Gulf Stream, South Wall Gulf Stream, and eddy list. The user may choose to run the previous analysis through the expert system prior to the labeling. This is done to progress the previous analysis to the time of the current image data. The function can accept at most eight input objects. The North Wall, if provided, is a single object. The South Wall, if provided, is a single object. Each eddy of an eddy list file, if provided, is an object. The user must input at least one of the previous analysis: North Wall, South Wall, or eddies.

The user is given several output options, at least one of which should be chosen. The first option is for a labeled output image file. This image file will contain pixels with specific values corresponding to the labeled objects. The values correspond to bit planes of the output image. The values for each type of label will be listed for the user. The user may, however, choose to have the three types of objects output separate, and a version of the edge image created less any or all of the labeled edges. The user is given the option of getting the North and South Walls as list files. These list files will be a regular SAMAS Gulf Stream list file. The first record will contain the number of positions. The positions will be latitude, longitude, and a source code of "O". The points in this file are not consecutive points along a Gulf Stream. The points are random Gulf Stream positions. The user is prompted for a pixel distance for creating these list files. This is to reduce the amount of data being saved. No points within the specified distance to a point saved in the output list will be kept. The user can choose to have just the eddy edges saved in an image file. Each eddy in the output file will have a different plane value associated with it. Finally, the user can have any of the labeled objects extracted from the input edge image to create an edge image with any or all of the North Wall, South Wall, or eddy edges missing (for more information on this type of labeling see Ref. 7). The prompts for creating the labeled output are:

Process or create shell script (def: Proc or Com) -

> The user is prompted for the type of run. If Proc (process) is selected, then the labeling will be executed as soon as all the inputs have been entered. If Com (command) is selected, then the user will be prompted for a shell script name and a shell script will be created. This shell script can be used for batch processing the creation of the labeled edges.

Name for shell script (def: Label.Com) -

If the user has selected to create a shell script, then this prompt is for the name of the shell script. A default name has been provided.

Input grey scale image file -

The user must enter the IR image used to create the edge image which is to be labeled.

Input edge image file -

The user must enter the input edge image to be labeled.

Input North wall list file (opt.) -

The user can choose to enter an optional input North Wall list file for the previous analysis to be used to label the North Wall edges of the input edge image. Prior to the labeling the user may have run this analysis through the Expert System to match the input image time.

Input South wall list file (opt.) -

The user can choose to enter an optional input South Wall list file for the previous analysis to be used to label the South Wall edges of the input edge image. Prior to the labeling the user may have run this analysis through the Expert System to match the input image time.

Input eddy list file (opt.) -

The user can choose to enter an optional input eddy list file for the previous analysis to be used to label the eddy edges of the input edge image. Prior to the labeling the user may have run this analysis through the Expert System to match the input image time.

Output labeled edge image file (opt.) -

The user may select as output a labeled edge image. This image contains the edges that were labeled as North Wall, South Wall, and eddies. Each object North Wall, South Wall, and each eddy has a unique plane value associated with it. Labels can only be created if an input analysis for that object was provided. The plane values for the objects will be listed for the user. This is an optional output.

Output North wall list file (opt.) -

The user can choose to have the North Wall extracted from the labeled edge image and output as a list file. This is an optional output file and the prompt for it will be given only if an input North Wall previous analysis was specified.

Pixel distance to separate Gulf Stream coding (def: 0) -

This prompt will be given only if an output North Wall list file is specified. The pixel distance is used to reduce the number of positions output to the above specified file. Once a position has been selected for the list, then no positions within the specified distance will be kept for output.

Output South wall list file (opt.) -

The user can choose to have the South Wall extracted from the labeled edge image and output as a list file. This is an optional output file and the prompt for it will be given only if an input South Wall previous analysis was specified.

Pixel distance to separate Gulf Stream coding (def: 0) -

This prompt will be given only if an output South Wall list file is specified. The pixel distance is used to reduce the number of positions output to the file. Once a position has been selected for the list, then no positions within the specified distance will be kept for output.

Output eddy image file (opt.) -

The user can choose to have the eddy edges extracted from the labeled edge image and output as an eddy image file. The image file will contain only the eddy edges, each labeled eddy will retain its unique plane value. This is an optional output file and the prompt for it will be given only if an input eddy previous analysis was specified.

Output frontal image file (opt.) -

The user can have any or all of the labeled edges masked from the input edge image to create a new frontal edge image.

Mask North wall from frontal image (Def: Yes) -

This prompt is given only if a frontal image file is to be produced and a North Wall previous analysis was specified. This prompt determines if the edges labeled as North Wall are to be masked from the original edge image.

Mask South wall from frontal image (Def: Yes) -

This prompt is given only if a frontal image file is to be produced and a South Wall previous analysis was specified. This prompt determines if the edges labeled as South Wall are to be masked from the original edge image.

Mask Eddies from frontal image (Def: Yes) -

This prompt is given only if a frontal image file is to be produced and an eddy previous analysis was specified. This prompt determines if the edges labeled as eddies are to be masked from the original edge image.

2.2.10 Label Edge Image (Topographic method)

The user can select this option from the menu or the command $SAMASEXE/label_driver2 can be used. The user may choose to run the functions used to create the labeled output or to create a shell script that can be used later to create the labeled output. Feature labeling is done by assigning topographic labels. This function creates an edge image internally. The idea for this method stems from fitting a bicubic polynomial to each pixel's neighborhood and assigning topological labels based on the first and second directional derivatives of the polynomial surface. The user supplies an IR image and a previous analysis for any or all of the following: North Wall Gulf Stream, South Wall Gulf Stream, and eddy list. The user may have run the previous analysis through the Expert System prior to this labeling. This is done to progress the previous analysis to the time of the current image data. The North Wall, if provided, is a single object. The South Wall, if provided, is a single object. Each eddy of an eddy list file is an object. The user must input at least one of the previous analysis: North Wall, South Wall, or eddies.

The user is given several output options, at least one of which should be chosen. The first option is for a labeled output image file. This image file will contain pixels with specific values corresponding to the labeled objects; these values are printed for the user. The user may choose to have the three types of objects output separate and a version of the edge image created less any or all of the labeled edges. The user is given the option of getting the North and South Walls as list files. These list files will be SAMAS Gulf Stream list files. The first record will contain the number of positions. The positions will be latitude, longitude, and a source code of "O". The points in this file are not consecutive points along a Gulf Stream. The points are random Gulf Stream positions. The user is prompted for a pixel distance for creating these list files. This is to reduce the amount of data being saved. No points within the specified distance to a point saved in the output list will be kept. The user can choose to have just the eddy edges saved in an image file. Each eddy in the output file will have a different value associated with it. Finally, the user can have any of the labeled objects extracted from the labeled edge image to create an edge image with any or all of the North Wall, South Wall, or eddy edges missing (for more information on this type of labeling see Ref. 2). The prompts for creating the labeled output are:

Process or create shell script (def: Proc or Com) -

> The user is prompted for the type of run. If Proc (process) is selected, then the labeling will be executed as soon as all the inputs have been entered. If Com (command) is selected, then the user will be prompted for a shell script name and a shell script will be created. This shell script can be used for batch processing the creation of the labeled edges.

Name for shell script (def: Label.Com) -

If the user has selected to create a shell script, then this prompt is for the name of the shell script. A default name has been provided.

Input grey scale image file -

The user must enter the IR image to be used for creating the labeled edge image.

Input North wall list file (opt.) -

The user can choose to enter an optional input North Wall list file for the previous analysis to be used to label the North Wall edges of the input image. Prior to the labeling the user may have run this analysis through the Expert System to match the input image time.

Input South wall list file (opt.) -

The user can choose to enter an optional input South Wall list file for the previous analysis to be used to label the South Wall edges of the input image. Prior to the labeling the user may have run this analysis through the Expert System to match the input image time.

Input eddy list file (opt.) -

The user can choose to enter an optional input eddy list file for the previous analysis to be used to label the eddy edges of the input image. Prior to the labeling the user may have run this analysis through the Expert System to match the input image time.

Output labeled edge image file (opt.) -

The user may select as output a labeled edge image. This image contains the edges that were labeled as North Wall, South Wall, and eddies. Each object North Wall, South Wall, and each eddy has a unique value associated with it. Labels can only be created if an input analysis for that object was provided. The values for the objects will be listed for the user. This is an optional output.

Output North wall list file (opt.) -

The user can choose to have the North Wall extracted from the labeled edge image and output as a list file. This is an optional output file and the prompt for it will be given only if an input North Wall previous analysis was specified.

Pixel distance to separate Gulf Stream coding (def: 0) -

This prompt will be given only if an output North Wall list file is specified. The pixel distance is used to reduce the number of positions output to the above specified file. Once a position has been selected for the list, then no positions within the specified distance will be kept for output.

Output South wall list file (opt.) -

The user can choose to have the South Wall extracted from the labeled edge image and output as a list file. This is an optional output file and the prompt for it will be given only if an input South Wall previous analysis was specified.

Pixel distance to separate Gulf Stream coding (def: 0) -

This prompt will be given only if an output South Wall list file is specified. The pixel distance is used to reduce the number of positions output to the file. Once a position has been selected for the list, then no positions within the specified distance will be kept for output.

Output eddy image file (opt.) -

The user can choose to have the eddy edges extracted from the labeled edge image and output as an eddy image file. The image file will contain only the eddy edges, each labeled eddy will retain its unique plane value. This is an optional output file and the prompt for it will be given only if an input eddy previous analysis was specified.

Output frontal image file (opt.) -

The user can have any or all of the labeled edges masked from the labeled edge image to create a new frontal edge image.

Mask North wall from frontal image (Def: Yes) -

This prompt is given only if a frontal image file is to be produced and a North Wall previous analysis was specified. This prompt determines if the edges labeled as North Wall are to be masked from the labeled edge image.

Mask South wall from frontal image (Def: Yes) -

This prompt is given only if a frontal image file is to be produced and a South Wall previous analysis was specified. This prompt determines if the edges labeled as South Wall are to be masked from the labeled edge image.

Mask Eddies from frontal image (Def: Yes) -

This prompt is given only if a frontal image file is to be produced and an eddy previous analysis was specified. This prompt determines if the edges labeled as eddies are to be masked from the labeled edge image.

2.2.11 Process Regions

The user can select this option from the menu or the command $SAMASEXE/rs5_driver can be used. This function accepts input SAMAS image files and segments them with an iterative histogram thresholding, split-merge region segmentor. The function computes descriptive parameters for the segmented image and allows the user to filter the results

to a more concise data set based on any of these descriptive parameters. The system provides the means to integrate region and edge detection results.

This function operates by applying a threshold to the image histogram at its medium level, then labeling contiguous, 4 connected groups with the same binary value with unique region numbers. At this point the segmenter merges regions based on size and intensity when the difference in size or intensity with one surrounding region is below a threshold value. This split/merge process continues for a number of passes as specified by the user. In every pass, the segmenter thresholds each group (region) at its median value to obtain the next subdivision.

Upon completion of the segmentation process, the segmentor tabulates regions and computes for every region the following parameters:

Region Size (pixels)
Average Intensity
Centroid X value (pixels)
Centroid Y value (pixels)
Aspect Ratio (circumference/size)
Internal Connectivity
Size Differential with Surrounding Regions (pixels)
Intensity Differential with Surrounding Regions (pixels)
Moment of Inertia (pixels)

The table of regions can be reduced by filtering it based on any one or more of these parameters. For example, if one is interested in all regions with an aspect ratio greater than 1.5, one can filter the table with parameters aspect ratio, value 1.5, and direction "up". The filtering process can be repeated indefinitely, and filtering can be based on the same parameter for multiple times.

If the user specified a labeled edge image, then this edge file is read and processed to identify the individual edge fragments in the image. The list of edge fragments is then correlated with the region data to generate groups of edge fragments that coincide spatially with common regions. For the success of this correlation process, it is important that the region segmentation and filtering steps yield regions that are neither very large nor very small. Edge/region correlation is a new technique and is intended for basic automated image analysis research. Trial and error should be used to find an optimal combination of region segmentation, region filtering, and edge/region correlation.

This function produces a SAMAS image file containing the segmented image and the segmentation table (if desired). The system also creates, at the user's request, a binary (1-byte/pixel) image file as well as a 2-byte/pixel file showing the region number of each pixel. If edge/region correlation was performed, then an edge/region correlation table showing the groups of edges identified is printed.

The prompts for this function are:

Select Type of run
1 for 512 x 512 input image
2 for 1024 x 1024 input image

Run Type -
Input image files are limited to the size 512 x 512 or 1024 x 1024. The user must select type of run based on the input image size.

Input image file -
The user supplies the name for the input SAMAS image file.

input edge file -
The user supplies the name of the input labeled edge image for processing.

input mask file -
The user supplies the input mask file which is used for masking areas of the image for no processing (i.e., clouds and land).

output image file -
The user specifies the name for the output SAMAS segmented image file.

Number of passes?
The user must specify the desired number of passes to be executed.

Filter segmentation table? [n]
The user specifies"'No" for do not filter or "Yes' for filter the segmented image.

Enter filter parameters interactive or from database? [i/d]
The user has the option of entering filter choices and parameters interactively, or these can be read from a user created file "filter.dat".

(1) size filter
(2) aspect ratio filter
(3) adjacent region av. intensity filter
(4) internal connectivity filter
(5) intensity filter
(6) adjacent region-number filter
enter number -
If interactive filtering is selected, then the user must select the filter type by number. Once all desired filtering options have been selected (this could be none) the user hits RETURN to stop filter prompt. For each selected filter option, the prompt below will be given.

-1 filters out all features LESS THAN CRITERION
+1 filters out all features GREATER THAN CRITERION
Enter direction (-1 or 1) and criterion

This prompt is given for all filter options selected. Criterion is defined by the filter selected.

write binary (1 byte/pixel; no header) image file [n]?
Output image file name?

The user can specify an optional output segmented file with a format of 1-byte/pixel row/column file.

write binary (2 bytes/pixel; no header) region label image file? [n]
Output image file name?

The user can specify an optional output segmented file with a format of 2-byte/pixel row/column file.

write segmentation table? [n]
Segmentation table name?

The user can choose to have the segmentation table printed.

2.2.12 Run CEOF - Gulf Stream Interpolation

The user can select this option from the menu or the command $SAMASEXE/ceof can be used. The user may choose to run the function or to create a shell script that can be used later for the same purpose. This function uses complex empirical orthogonal functions to create a continuous Gulf Stream North Wall from fragmented segments and positions. The output Gulf Stream North Wall from the labeling is usually used as the input to be interpolated. The input Gulf Stream can be entered as an image file along with the range of values for pixel intensities or planes required to extract the positions; or an actual Gulf Stream list file can be entered. If an input Gulf Stream list file is used as the input to be interpolated, then the code values associated with the positions are used to determine weighting. Positions with codes of "A" for altimeter are weighted higher than positions with other code values. If an image file is used, then all positions are weighted equal. Also, input to this function is either a mode file containing the initial mode coefficients or a good Gulf Stream list file; one that gives a realistic description of a previous Gulf Stream. The user can provide a good quality previous analysis Gulf Stream as input to the INTERP option and a mode file will be created, or a mode file from a previous run of this function can be used. The user must also choose the number of eigenvectors to be used for the interpolation. Output from this function is a mode file and an interpolated continuous Gulf Stream created from the input partial Gulf Stream. For more information on the CEOF interpolation see Ref. 8. The prompts for this function are:

Process or create shell script (def: Proc or Com) -

The user is prompted for the type of run. If Proc (process) is selected, then the function will be executed as soon as all the inputs have been entered. If Com (command) is selected, then the user will be prompted for a shell script name and a shell script will be created. This shell script can be used for batch processing the creation of the interpolated Gulf Stream and associated mode file.

Name for shell script (def: CEOF.Com) -

If the user has selected to create a shell script, then this prompt is for the name of the shell script. A default name has been provided.

Input labeled Gulf Stream -

The user must enter the Gulf Stream that is to be used for the interpolation. This Gulf Stream can be partial and need not be continuous. The file can be in the form of an image or a Gulf Stream list. If the file is an image file, then the user will be prompted for type of values, the range for the values, and a spacing increment.

Input file type (Def: List or Image) -

The user must define the type of input Gulf Stream file list or image. If the input file is image, then the user will need to answer the next three prompts, else they are skipped.

Type of extraction (Def: Values or Planes) -

The user must state whether the positions within the image file are intensity values or plane values. This information is used for extracting the Gulf Stream positions from the image.

Value limits for extraction (Def: 1 1) -

Plane limits for extraction (Def: 1 1) -

The user must supply the value or plane ranges minimum and maximum at one of these prompts for extracting the Gulf Stream positions from the input image file.

Minimum pixel distance between points (Def: 0) -

The user may choose to reduce the number of Gulf Stream positions extracted from the image file. No pixel position within this specified distance of a pixel already selected for the Gulf Steam list will be saved.

Run INTERP (Def:Yes or No) -

The user has the option of running INTERP on an input Gulf Stream list to create a mode file to be used to start the interpolation. If INTERP is run, then the user will be prompted for an input Gulf Stream file. If INTERP is not run, then the user will be prompted for an input mode file.

Input Gulf Stream for INTERP to create mode file -

The user must input a realistic previous Gulf Stream list file to be used to create a mode file for OPTO.

Input Mode file for Opto -

The user must enter a mode file to be used to start the interpolation process. This file can be the mode file output of a previous run of this function.

Output Gulf Stream File -

The user must provide the name for the new Gulf Stream list file.

Output Mode File -

The user must provide the name for the mode file associated with the new Gulf Stream list.

Number of eigenvectors to be used for fitting (Def: 10) -

The user is prompted for the number of eigenvectors to be used for the interpolation of the Gulf Stream.

2.2.13 Run Hough - Eddy Detect

The user can select this option from the menu or the command $SAMASEXE/hough can be used. The user may choose to run the function or to create a shell script that can be used later for the same purpose. Input to this function is a labeled edge image. The image may contain eddies only or be of mixed labels. If the image is of mixed labels, then the user will be required to supply the type of values (pixel intensity or planes) and a range for extracting just the eddy edges. The user is also allowed an option for dilating the edges prior to the eddy detection.

Given a single radius the Hough Transform is applied as follows. An accumulator array is created. This array will be the same size as the image array and will be initially set to zeros. The image array is scanned for edge pixels. When an edge pixel is found, then all entries in the accumulator array are incremented which are the radius distance from that pixel's corresponding accumulator coordinates. These entries represent the center coordinates of all circles of the given radius that would contain on their boundary that edge pixel.

The user may have selected a width greater than one. In this case the Hough Transform will be applied for a donut or set of concentric circles. The minimum and maximum radius values are converted from kilometers to pixels. For each radius from minimum radius to maximum radius, incrementing by the radius increment, the accumulator array will be set to zero, the Hough Transform applied for that radius or set of radii; the entries selected from the accumulator array that meet the threshold test will be stored in a list. The list

will contain the radius, center coordinates, and normalized accumulator value. When width is greater than one, the Hough Transform is applied multiple times without resetting the accumulator array to zeros. Width defines the number of radii for which the Hough Transform will be applied without resetting the accumulator array to zeros. The user is given a parameter for type of weighting. This parameter is used to define weights for the radii. If type of weighting is "Ones", then all radii are weighted equal. If type of weighting is "Decrease", then the smaller radius is weighted by one incrementing by one up to the largest radius of the donut. If type of weighting is "Increase", then the larger radius is weighted by one incrementing by one to the smallest radius of the donut. The radius that will be stored in the list will be the larger radius of the donut. The accumulator entries are normalized so that they can be tested against a percentage threshold. Given the number of pixels for each of the radius and the desired weighting for each of the radius, a maximum possible accumulator value is computed. The computed accumulator values are divided by this maximum possible value. This normalized value is then compared with the percentage threshold that was specified. The entries, which are equal or greater than the threshold, are stored in a list for further processing.

After the Hough Transform has been applied for all the desired cases, the list of entries that passed the threshold test is sorted. The entries are sorted first in descending order by the radius and second in descending order by the normalized accumulator values. The function is biased to the larger eddies. Each entry is checked against the preceding entries in the list. A circle will be eliminated from the list if it is found to overlap another circle defined in a previous entry in the list.

The edited list of center image coordinates, pixel radius, and normalized accumulator values must be converted to a list of output eddies. The image sample, line coordinates are converted to latitude, longitude coordinates. The pixel radius is converted to kilometers. The center latitude, center longitude, kilometer radius, type code (U for Undefined) and source code (ED) are all written to the output eddy list file.

The prompts for eddy detection are as follows:

Process or create shell script (def: Proc or Com) -

> The user is prompted for the type of run. If Proc (process) is selected, then the function will be executed as soon as all the inputs have been entered. If Com (command) is selected, then the user will be prompted for a shell script name and a shell script will be created. This shell script can be used for batch processing the eddy detection.

Name for shell script (def: Hough.Com) -

> If the user has selected to create a shell script, then this prompt is for the name of the shell script. A default name has been provided.

Input edge image -

The user must input the labeled image containing the eddy edges. This image may contain eddy edges only or contain mixed edges. The user must state which case applies in the next prompt.

Input file type (Def: Eddies or Mixed) -

The user must state whether the eddy image file contains only eddy edges or also contains other types of edges, If the image is of mixed labels, then the user must answer the next two prompts to define the edges to be considered as eddy edges.

Type of extraction (Def: Values or Planes) -

The user must specify whether the eddy edges are to be extracted by intensity value or plane values. This is used for mixed label input images only.

Value limits for extraction (Def: 1 1) -

Plane limits for extraction (Def: 1 1) -

The user must define the data ranges for extracting the eddy labels from the input mixed label image. The prompt will be for intensity values or planes depending on the previous parameter.

Dilate eddy edges (Def: Yes)? -

The user can choose to have the input eddy edges dilated. Dilated edges usually work better for eddy detection, since eddies are not truly circular.

Number of desired dilate steps (def: 1) -

If the edges are to be dilated, then the user must specify how much dilation is desired. For a single pass of dilation the eight neighbors of any edge pixel are set to be edge pixels. Single pixel edges will become three pixels wide for a single pass of dilation.

Output eddy list file -

The user must provide the output eddy file name. If no eddies are found, then no file will be created and a message will be given.

Minimum and maximum radius to consider (Km) (Def: 50.000000 133.000000) -

The user must specify the minimum and maximum radius values to be processed. This value is in kilometers.

Circle mask pixel width (Def: 1) -

The user must enter the width or number of pixel radii to be used in detecting circles for each increment of radius.

Radius check pixel increment (Def: 1) -

The user may select not to test for all radii between the selected minimum and maximum radius values. An increment of two means every other radius will be tested.

Type of weighting ones, decrease to center, or increase to center (Def: O*nes, D*ecrease, I*ncrease) -

This parameter is only meaningful if the width value is greater than one. When using multiple radii for detecting circles the concentric circles forming the donut can all be weighted equal "Ones", the outer radii can be weighted higher than the inner radii "Decrease", or the output radii can be weighted less than the inner radii "Increase". The range of weights for the decrease and increase is one incrementing by one for each new radii of the donut.

Threshold for circles (percentage 0 - 100) (Def: 40.000000) -

This threshold is used to define which circles will be kept in the list for sorting. No circles with a normalized accumulator value less than this threshold will be considered. The weights are normalized by taking the weighted sum of points of the donut or single radius and dividing by the maximum possible sum.

2.2.14 Apply Image Mask

The user can select this option from the menu or the command $SAMASEXE/mask can be used. This function is used to apply a mask to an image file. The user enters the image to be masked and one or more images to be used as masks. The user selects the option of masking the zero or nonzero values of the mask images. If the zero values of the mask images are to be used, then for all zero values of the input mask images the constant value specified by the user will be used as the output value, else the value from the background image will be used as the output value. If the nonzero values of the mask images are to be used, then for each nonzero value of the input mask images the constant value specified by the user will be used as the output value, else the value from the background image will be used as the output value. The prompts for this function are:

Background image -

The user must specify the background image. The nonmasked values of the output image will have the background image values.

Mask image -

This prompt will be given multiple times until no name is entered. The user can enter one or more images to be used as masks. Either the zero or nonzero values of these images will be set to a constant value in the output image.

1 - Mask with constant for non-zero values of mask
2 - Mask with constant for zero values of mask
Select option from list (Def: 1) -

The user must select whether the zero or nonzero values of the input mask images are to be used.

Input constant value to be used (Def: 0) -

The user must select the constant value to be used for the masked pixels of the output image.

Output image -

The user must supply the name for the output image file. The data type and mapping information will be the same as for the input background image.

2.2.15 Convert Gulf Stream for Connect

The user can select this option from the menu or the command $SAMASEXE/gs_code_l can be used. The purpose of this function is to read a Gulf Stream list file, change all codes to "l", and write the list to a new Gulf Stream list file. When the new Gulf Stream is plotted for viewing by the editor, all points will be connected. The program prompts are:

Input File -

The user must specify an input Gulf Stream list file name.

Output File -

The user must specify an output Gulf Stream list file name.

2.2.16 System Command

When the user selects this option from the menu a prompt will be given for the user to enter a regular UNIX command. This allows the user to execute a shell script, list a directory, etc. Some UNIX commands do not work in this mode, example: cd.

2.2.17 Change Directory

When the user selects this option from the menu a prompt will be given for the user to enter the desired directory. This will become the default directory until it is changed or the user elects to exit PV-WAVE.

3.0 SAMAS Standard Input/Output Files

The standard input and output files for the SAMAS are Gulf Stream list, eddy list, and image. These files have defined formats. All modules developed for the system will use these formats for these types of files.

3.1 Gulf Stream List File Format

The Gulf Stream list files are ASCII. The file names should end in the suffix ".gs". The first record contains the number of positions contained in the file. The remaining records contain the Gulf Stream positions--latitude, longitude, and a source code. The source code values that have been established thus far are "I" for image, "A" for altimeter, and "O" for other. An example of other would be CEOF interpolated positions. Consecutive positions with code values of I are expected to be consecutive connected points of the Gulf Stream list. The format for these records is:

Header (Record 1)

I5	Number of positions contained in the file

Positions (Record 2 - EOF)

F10.4	Latitude	(+ North, - South)
F10.4	Longitude	(+ East, - West)
A2	Source Code	(I image, A altimeter, O other; preceded with a single blank)

3.2 Eddy List File Format

The eddy list files are ASCII. The file names should end in the suffix ".eddy". The first record contains the number of eddies contained in the file. The remaining records contain the eddy information, one eddy per record. The eddy information consists of center latitude, center longitude, kilometer radius, eddy type (Warm, Cold, or Undefined), and a source code. The source code is for the analyst to trace the origin of an eddy. Some functions, such as the Expert System, assign source codes for output eddies. The source code output by the Expert System is "ES". The format for these records is:

Header (Record 1)

I5	Number of eddies contained in the file

Eddy (Record 2 - EOF)

F10.4	Center Latitude	(+ North, - South)
F10.4	Center Longitude	(+ East, - West)
F10.4	Radius	(Kilometers)
A2	Eddy Type	(W warm, C cold, U undefined; precede with a single blank)
A3	Source Code	(2-character code; precede with a single blank)

3.3 Image File Format

SAMAS uses the image file format designed for NSIPS. The file names should end with the suffix ".img", and start with the prefix "w_". Code is available for reading and writing these data files; see section 4.1 below. The header format is shown in Appendix B. Some modules allow output of row/column files. These files should end with suffix ".rc". These files have no header record or record delimiters. Data can be byte, int*2, int*4, or float.

4.0 SAMAS Common Source

There is some source code that has been developed for use by many of the SAMAS functions. These are for image I/O and map registration of image data.

Several of the functions listed in the SAMAS menu require multiple executable files. For these functions driver programs have been created. These driver programs create shell scripts that execute the modules after getting the user input or, in some cases where specified by the user, can be used to create shell scripts for later use. Code has been developed for these driver programs to get user input values and removing (deleting) working files created by the program modules.

4.1 Image I/O

The source code used to read and write the image file data is found in:

$SRCU/image_io.f
$SRCC/io_suntran.c

To use the standard image I/O routine the user must compile and link the above source code with their calling routine and call the Fortran subroutine imageio with the appropriate parameters. This routine is used to open, close, read, and write an image file. A description of the parameters is:

imageio (File_Number, Mode, IO_Error, Buffer, Buffer_Type, Line_Number, File_Type, Number_Samples, Number_Lines, North, West, Degree_Pixel, Projection)

File_Number	Int*4	Fortran unit number associated with the opened file
Mode	Int*4	Defines the type of operation to be performed by imageio 1 - Open file and read header information 2 - Open file and write header information 3 - Read the specified line of image data 4 - Read the next line of image data 5 - Write the next line of image data 6 - Close the image file
IO_Error	Int*4	IOstat value returned from the Fortran open, read, write, or close
Buffer	Int*4	Contains the image file name for open and contains a line of data for read and write operations
Buffer_Type	??	Defines the type of data in the buffer being sent for a write or returned by a read (1 Byte, 2 Int*2, 3 Int*4, 4 Real)
Line_Number	Int*4	Used when reading a specific line of data from the image file
File_Type	Int*4	Defines the type of image data in the file (1 Byte, 2 Int*2, 3 Int*4, 4 Real)
Number_Samples	Int*4	Number of samples contained in a line of image data
Number_Lines	Int*4	Number of lines of image data in the file
North	Real*4	North limit for the image map registration
West	Real*4	West limit for the image map registration
Degree_Pixel	Real*4	Degree/pixel ratio for the image map registration

imageio Mode Values and Arguments						
Mode → Value Argu-ments ↓	1 Open Read	2 Open Write	3 Read Select Record	4 Read Next Record	5 Write Next Record	6 Close
File_ Number	Return Integer	Return Integer	Pass Integer	Pass Integer	Pass Integer	Pass Integer
Mode	Pass Integer	Pass Integer	Pass Intege r	Pass Integer	Pass Integer	Pass Integer
IO_ Error	Return Integer	Return Integer	Return Integer	Return Integer	Return Integer	Return Integer
Buffer	Pass Integer	Pass Integer	Return *	Return *	Pass *	Not Used
Buffer_ Type	Not Used	Not Used	Pass Integer	Pass Integer	Pass Integer	Not Used
Line_ Number	Not Used	Not Used	Pass Integer	Not Used	Not Used	Not Used
File_ Type	Return Integer	Pass Integer	Not Used	Not Used	Not Used	Not Used
Number_ Samples	Return Integer	Pass Integer	Not Used	Not Used	Not Used	Not Used
Number_ Lines	Return Integer	Pass Integer	Not Used	Not Used	Not Used	Not Used
North	Return Real	Pass Real	Not Used	Not Used	Not Used	Not Used
West	Return Real	Pass Real	Not Used	Not Used	Not Used	Not Used
Degree_ Pixel	Return Real	Pass Real	Not Used	Not Used	Not Used	Not Used
Projec tion	Return Integer	Pass Integer	Not Used	Not Used	Not Used	Not Used

Table 1. Mode values and arguments for imageio routine.

Projection	Int*4	Type of map projection (0 rectilinear, 1 mercator)

See Table 1 for how to use the arguments for the different mode values.

When output image files are closed this functions spawns an executable file, which is part of the NSIPS software. The purpose for this is to compute and store some of the values contained in the image header record. The required executable is $SAMASEXE/mk_rc_to_img (copied from /sips3d2/run/imagepro/Bin/Sun/mk_rc_to_img).

4.2 Map Registration

Each image data file contains as part of the header record information defining the map registration for that image. These parameters include the North limit, West limit, degree/pixel ratio, and map projection type (rectilinear, mercator, etc.). Some of these values are written to the image header when the file is opened for write, other values are computed and written when the file is closed. Routines are provided to do the following convert sample, line positions to latitude, longitude positions; convert latitude, longitude positions to sample, line positions; convert pixel distance to kilometer distance; convert kilometer distance to pixel distance; and compute the kilometer distance between two latitude, longitude positions. The source code is located in:

$SRCU/mapf.f
$SRCU/mapc.c

The mapf.f file includes:

$SRCF/xy_00_ll.f
$SRCF/xy_01_ll.f
$SRCF/xy_02_ll.f
$SRCF/xy_03_ll.f
$SRCF/xy_04_ll.f
$SRCF/xy_05_ll.f
$SRCF/xy_06_ll.f
$SRCF/xy_tm_ll.f
$SRCF/xy_to_ll.f
$SRCF/ll_00_xy.f
$SRCF/ll_01_xy.f
$SRCF/ll_02_xy.f
$SRCF/ll_03_xy.f
$SRCF/ll_04_xy.f
$SRCF/ll_05_xy.f
$SRCF/ll_06_xy.f
$SRCF/ll_tm_xy.f
$SRCF/ll_to_xy.f
$SRCF/coord_conv.f

$SRCF/mapfuncts.f
$SRCF/fourier.f
$SRCF/ind_fourier.f
$SRCU/image_io.f
$SRCU/map_xy_ll.f

The mapc.c file includes:

$SRCC/io_suntran.c

The routines to be called for mapping are all written in Fortran. To use these subroutines for computing latitude, longitude, sample, line, etc., the map array must first be initialized. The SAMAS image file must be opened. The Fortran unit file number must be supplied to the initialize routine. The initialize routine reads the image header record and initializes the mapping array required for all coordinate conversions. The call to initialize the map array is:

map_init(fnum, Relative, Map_array)

int*4	fnum (*)	Passed	Fortran file unit number for open, read, write, close file
int*4	Relative	Passed	Sample, line values will be 0 relative if set to 0 or 1 relative if set to 1
real*8	Map_array(21)	Returned	Mapping array

Once the mapping array has been initialized, then it can be used for computing the coordinate conversions and distances. The following call computes the latitude, longitude position from the X, Y position:

cvt_xy_to_ll(Lat, Lon, X, Y, Map_array)

real	Lat	Returned	The computed latitude value
real	Lon	Returned	The computed longitude value
int*4	X	Passed	X is the sample position, 0 or 1 relative depending on the value of Relative in the initialization of Map_Array

int*4	Y	Passed	Y is the line position, 0 or 1 relative depending on the value of Relative in the initialization of Map_Array
real*8	Map_array(21)	Passed	Map array returned from map_init

The following call computes the X,Y position from a latitude, longitude position:

cvt_ll_to_xy(Lat, Lon, X, Y, Map_array)

real	Lat	Passed	The latitude position
real	Lon	Passed	The longitude position
int*4	X	Returned	Computed X or line position is 0 or 1 relative depending on the value of Relative in the initialization of Map_Array
int*4	Y	Returned	Computed Y or line position is 0 or 1 relative depending on the value of Relative in the initialization of Map_array
real*8	Map_array(21)	Passed	Map array returned by map_init

Images are mapped using a degree/pixel ratio. The relationship of kilometers to degrees varies with latitude and differs for latitude and longitude. A routine has been developed to convert a pixel distance to kilometer distance. The routine computes kilometer distance from the given X, Y position to due South the specified pixel distance and from the given X, Y position to due east the specified pixel distance. These two distances are averaged to get the returned kilometer distance. The following call converts the pixel distance at the specified X, Y position to kilometers:

pixels_to_km(X, Y, Pdis, KMdis, Map_array)

int*4	X	Passed	X position for start of pixel distance
int*4	Y	Passed	Y position for start of pixel distance

int*4	Pdis	Passed	Pixel distance to be converted to kilometers
real*4	KMdis	Returned	Computed kilometer distance for the pixel distance
real*8	Map_array(21)	Passed	Map array returned by map_Init

The conversion of a kilometer distance to a pixel distance is done by the same principle as that described above. Now the position is specified as latitude, longitude and the conversion is from kilometers to pixels. The following call computes the kilometer distance centered at the specified latitude, longitude position to pixels:

km_to_pixels(Lat, Lon, KMdis, Pdis, Map_array)

real	Lat	Passed	Latitude position for start of kilometer distance
real	Lon	Passed	Longitude position for start of kilometer distance
real	KMdis	Passed	Kilometer distance to be converted to pixel distance
int*4	Pdis	Returned	Pixel distance computed for the kilometers distance
real*8	Map_array(21)	Passed	Map array returned by map_init

The last routine provided for mapping computes a kilometers distance between two latitude, longitude positions. This is not image dependent and thus does not require that Map_Init be run. The following call will compute a kilometer distance between two latitude, longitude positions:

km_dist(Dis, Lon1, Lon2, Lat1, Lat2)

real*4	Dis	Returned	Computed distance in kilometers
real*4	Lon1	Passed	Longitude for the first position
real*4	Lon2	Passed	Longitude for the second position
real*4	Lat1	Passed	Latitude for the first position

real*4	Lat2	Passed	Latitude for the second position

4.3 Driver Input from User

Code is provided for getting real, integer*4, and character data from the user. The source for this code is in:

$SRCU/input.c

There are three subroutines, one for each of the three types of data--real, integer, and character. These routines are written in C. Each routine will allow the user to hit RETURN to indicate that the defaults or no input is desired. The call for the function to input character data is:

void InputChar(Input, Prompt)

char	Input[]	Returned	Contains string input from the user. If no string is entered, then no change is made to the buffer
char	Prompt[]	Passes	String used to prompt the user for input

The call for the function to input float or real data is:

void InputFloat(Prompt, Nval, Val, DefVal)

char	Prompt[]	Passed	String used to prompt the user for input
int	Nval	Returned	Number of values entered by the user, if no values are entered by the user, then this will be set equal to DefVal
float	Val[]	Returned	Values entered by the user, if no values are entered, then no change is made to this argument
int	DefVal	Passed	Value to be assigned to Nval if no values are entered by the user, should denote the number of default values contained in Val

The call for the function to input integer data is:

void InputInt(Prompt, Nval, Val, DefVal)

char	Prompt[]	Passed	String used to prompt the user for input
int	Nval	Returned	Number of values entered by the user, if no values are entered by the user, then this will be set equal to DefVal
int	Val[]	Returned	Values entered by the user, if no values are entered, then no change is made to this argument
int	DefVal	Passed	Value to be assigned to Nval if no values are entered by the user, should denote the number of default values contained in Val

4.4 Remove File

An executable module has been provided to remove (or delete) files. This is used by some of the drivers to delete intermediate files created by one module of a function to be used by another module of the same function. The source for this module is:

$SRCU/remove.c

To use the executable as a UNIX type command use the executable:

$SAMASEXE/remove fname

Where fname is the file to be deleted.

5.0 Discussion

The SAMAS version 1.1 is designed for easy growth and change. As new modules are created they can be easily added to the menu command list as new options. As better methods are developed for some of the existing modules, replacement is made easy by replacing records in the menu command list.

For a listing of functions and their required files see Appendix D. Some options have driver routines that spawn other executables. The executables, source for all executables, and data files for each function are listed. The files used to compile the system are also shown in Appendix D. The Expert System uses a make file. The remaining functions use shell script files to accomplish this task.

6.0 Acknowledgments

SAMAS 1.2 has been made possible by the hard work of many individuals. The system was supported by the Office of Naval Technology, Program Element 0602435N, Program Manager Dr. Thomas Warfield. Major modules of SAMAS 1.2 were developed under the Data Assimilation and Analysis Technology Project of this program element, Ronald Holyer Project Manager. The Expert System was developed by Perceptics Corporation/University of Tennessee, Mathew Lybanon contract technical officer (Contract No. N00014-87-C-6001). The labeling software was developed by Louisiana State University, Sankar Krishnamurthy, N. Krishnakumer, and S. Sitharama Iyengar (Contract No. N00014-88-K-6002). Planning Systems Incorporated developed the Complex Empirical Orthogonal Function interpolation of the Gulf Stream, Eugene J. Molinelli and Michael J. Flanigan (Contract No. N66604-86-D-0120). Bobby Grant of Sverdrup Technology, Inc. provided much of the processing for the testing and validation of the SAMAS 1.2.

7.0 References

[1] Susan Bridges (1992). *Adding Explanation Capability to a Knowledge-Based System for Interpretation of Oceanographic Images*, Naval Research Laboratory, NOARL Tech. Note 296.

[2] Sankar Krishnamurthy, S. Sitharama Iyengar, Ronald Holyer, and Matthew Lybanon (1992). Topographic Based Feature Labeling for Infrared Oceanographic Images, submitted to: *Pattern Recognition Letters.*

[3] M. Lybanon, J.D. McKendric, R.E. Blake, J.R.B. Cockett, and M.G. Thomason (1986). A Prototype Knowledge-Based System to Aid the Oceanographic Image Analyst, *SPIE Vol 635-Applications of Artificial Intelligence III*, pp 203-206.

[4] R.J. Holyer and S.H. Peckinpaugh (1989). Edge Detection Applied to Satellite Imagery of the Oceans, *IEEE Trans. Geosci. Remote Sensing* 27, pp 46-56.

[5] Sarah H. Peckinpaugh (1991). An Improved Method for Computing Gray-Level Cooccurrence Matrix Based Texture Measures, *CVGIP: Graphical Models and Image Processing* vol 53 no 6, pp 574-579.

[6] Theo Pavlidis (1980). A Thinning Algorithm for Discrete Binary Images, *Computer Graphics and Image Processing* 13, pp 142-157.

[7] N. Krishnakumar, S. Sitharama Iyengar, Ronald Holyer, and Matthew Lybanon (1990). Feature Labelling in Infrared Oceanographic Images, *Image and Vision Computing* vol 8 no 2.

[8] E.J. Molinelli and M.J. Flanigan (1987). *Optimized CEOF Interpolation of the Gulf Stream*, Planning Systems Inc., Tech. Rept. TR-392395, Contr. N66604-86-D-0120.

[9] Juanita R. Chase and Ronald J. Holyer (1993). Neural Networks for Gulf Stream Dynamics, *SPIE : Applications of Artificial Neural Networks IV,* Vol. 1965, pp 286-297.

[10] *User's Guide for the Naval Research Laboratory Satellite Image Processing System (NSIPS)* to be published as a Memorandum Report.

Appendix A

USER'S GUIDE

FOR THE

SEMI-AUTOMATED MESOSCALE ANALYSIS SYSTEM
(SAMAS)

Prepared For:

Remote Sensing Branch
Ocean Sensing and Prediction Division
Naval Research Laboratory
Stennis Space Center, MS 39529

Prepared By:

Scientific Systems Section
Sverdrup Technology, Inc.
SSC Technical Support Service Contractor
Stennis Space Center, MS 39529

September 1992

TABLE OF CONTENTS

2.4.4.2 Trace Gulf Stream

2.4.4.2.1 Delete to Beginning of Line

2.4.4.2.2 Delete to End of Line

2.4.4.2.3 Add to Beginning of Line

2.4.4.2.4 Add to End of Line

2.4.4.2.5 Add Front Trace to Beginning of Line

2.4.4.2.6 Add Front Trace to End of Line

2.4.4.2.7 Connect All Points

2.4.4.2.8 Unconnect Points in Section of Line

2.4.4.2.9 Redraw Middle of Line

2.4.4.2.10 Trace Help

2.4.5 Save a Gulf Stream File

2.4.6 Load a Gulf Stream File

2.4.7 Label Gulf Streams

2.4.8 Change Default Color

2.4.9 Change Default Thickness

2.4.10 Change Default Style

2.4.11 Gulf Stream Editor Help

2.5 Eddy Edit

2.5.1 Draw Eddy Using Fixed Center

2.5.2 Draw Eddy Using Fixed Radius

2.5.3 Draw Eddy Using Two Fixed Points

2.5.4 Enter New Eddy

2.5.5 Modify an Eddy

2.5.6 Delete an Eddy

2.5.7 Print Eddy Info

2.5.8 Save an Eddy File

2.5.9 Load an Eddy File

2.5.10 Label Eddies

2.5.11 Change Default Color

2.5.12 Change Default Thickness

2.5.13 Change Default Style

2.5.14 Eddy Editor Help

2.6 Write a Picture File

2.7 Increase Magnification

2.8 Decrease Magnification

2.9 Annotate

2.9.1 Annotate

2.9.2 Remove an Annotation

2.9.3 Change Color

2.9.4 Change Size

2.9.5 Change Text

2.9.6 Change Orientation

2.9.7 Change Background Color

LIST OF FIGURES

USER'S GUIDE

SEMI-AUTOMATED MESOSCALE ANALYSIS SYSTEM (SAMAS)

1.0 INTRODUCTION

The purpose of this document is to describe the capabilities of the Semi-Automated Mesoscale Analysis System (SAMAS) which is an added feature to the NRL Satellite Image Processing System (NSIPS). The capabilities of the NSIPS software are described in the NSIPS Users Guide. The SAMAS software was written to run on a SUN SPARC station under the Precision Visuals-Workstation Analysis and Visualization Environment (PV-WAVE) command language.

The purpose of the SAMAS software is to provide users with the capability to display images and indicate gulf streams, eddies, and fronts within them. This software is composed of PV-WAVE, FORTRAN, and C source code.

The SAMAS software is a point and click package with most of the user inputs being made by indicating the desired choice with the mouse pointer and clicking the indicated mouse button. Variable inputs, such as the name of a new file, are prompted for in the window labeled "MAIN". When inputs are requested in this manner, the shell tool from which NSIPS was started must be selected or the input will not be read. Outputs and errors will be displayed in pop-up windows, the MAIN window, and in the shell tool from which NSIPS was started.

1.1 System Overview

The SAMAS software has three main components and several auxiliary functions. The main components of the SAMAS software are: 1. Frontal Editor; 2. Eddy Editor; and 3. Gulf Stream Editor. The SAMAS software is activated from the "User Functions" menu of NSIPS.

1.3 System Limitations

The SAMAS software is not a stand alone package. It must be run as part of the NSIPS software. Also, it is necessary for the shell tool from which the NSIPS software was started to be selected in order to enter input from the keyboard.

2.0 Detailed Description

This section describes the functions of the SAMAS software. Upon entering the SAMAS function, the software checks for any images on the screen. If screen images are present, the user is given the option of working with an existing warped screen image or loading a warped image file. If no screen images are present, the user is presented a list of available warped image files that may be worked with. When loading a warped image file, the user is prompted for scaling method, display size and whether or not to overlay a map, grid, and land mask. The "SAMAS Main Menu", shown in Figure 2.0-1, is displayed after the working image has been selected. The functions available from the "SAMAS Main Menu" are described in the sections that follow.

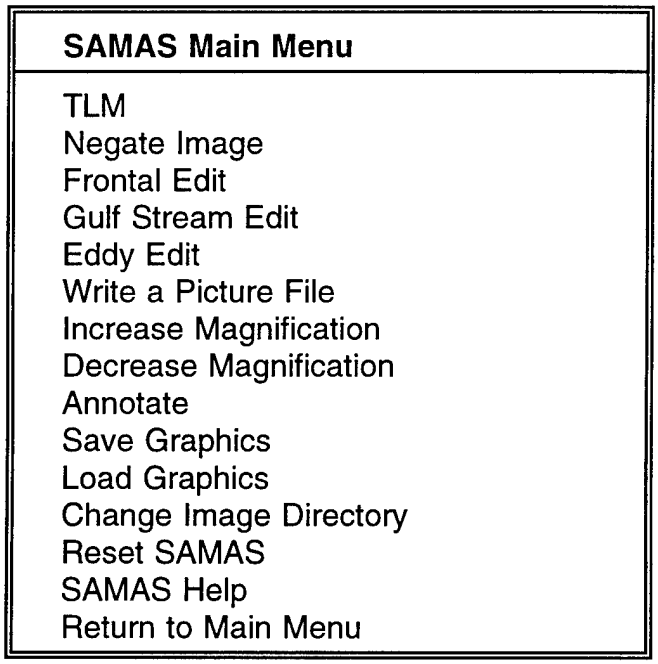


Figure 2.0-1 SAMAS Main Menu

2.1 TLM

The "TLM" function allows the user to interactively change the color table with the mouse. A new window is created and a graph of the color output value versus pixel value is displayed and modified interactively. This function has three different modes of operation which are selected by clicking the left mouse button while the mouse pointer is on the mode name and another mode is not engaged. The modes are exited by clicking the right mouse button. Clicking the right mouse button twice exits

the function. The three modes of operation for the "TLM" function are "Ramp", "Segments", and "Draw". The "Ramp" mode allows the user to select the left endpoint with the left mouse button and the right endpoint with the middle mouse button. The cursor is moved along the axis border with either button depressed for easiest control. The "Segments" mode allows the user to adjust the color table in segments. The left mouse button is used to start a new segment and the middle mouse button is used to mark additional vertices and continue the segment. The "Draw" mode allows the user to mark a series of points by depressing the left mouse button while moving the mouse pointer. Releasing the button updates the color table. There is a help screen available for this feature.

2.2 Negate Image

The "Negate Image" function allows the user to invert the color values in the SAMAS working image without adjusting the color table. Selecting this function again will return the working image back to its original values.

2.3 Frontal Edit

The "Frontal Edit" function allows the user to load, edit, and save frontal image files. Selecting this function brings up the "Frontal Editor Menu" shown in Figure 2.3-1.

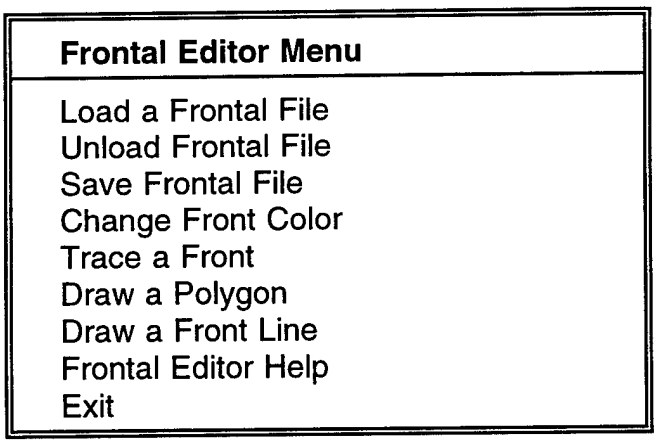


Figure 2.3-1 Frontal Editor Menu

2.3.1 Load a Frontal File

The "Load a Frontal File" function of the Frontal Editor allows the user to load a file containing front lines and display it overlaid on the working image. Frontal files are warped image files with names in the format of "w_<generic_name>front.img". They must be of the same size and for the same geographic location as the current working

image and must be of the data type byte. The header format for warped image files is given in Appendix II. When a frontal file is loaded, the display size is automatically adjusted in the same manner as the working image. The presence of a frontal line element is indicated by a pixel intensity value of 1 in the frontal file. All other pixel intensity values are 0. Loading a frontal file with one already displayed automatically replaces the existing front lines with the front lines in the file being loaded.

2.3.2 Unload Frontal File

The "Unload Frontal File" function of the Frontal Editor allows the user to remove all frontal lines from the working image.

2.3.3 Save Frontal File

The "Save Frontal File" function of the Frontal Editor allows the user to save the currently displayed frontal lines to a warped image file with a name of the format "w_<root_name>front.img". The root name is prompted for and entered through the main window. If a file name is selected that already exists the system will prompt the user to ensure the file should be overwritten. The header information and the output size of the frontal image are automatically adjusted to match the warped image file which was read in as the working image. The data type of the frontal image is always byte. The header format for warped image files is given in Appendix II.

2.3.4 Change Front Color

The "Change Front Color" function of the Frontal Editor allows the user to change the color in which all front lines are displayed from a list of ten standard colors, see Appendix III. The new color applies to all existing and new front lines.

2.3.5 Trace a Front

The "Trace a Front" function of the Frontal Editor allows the user to select a group of connected front lines for deletion. The user selects the front lines by placing the mouse pointer onto the desired front lines (or within five pixels) and clicking the left mouse button. Clicking the right mouse button will exit the function. When the frontal line has been selected, all front lines connected to that frontal line will cycle through the ten standard colors and the user will be prompted to keep or delete the selected lines. Clicking the left or middle mouse buttons exits the function keeping the selected frontal lines. Clicking the right mouse button exits the function deleting the selected frontal lines.

2.3.6 Draw a Polygon

The "Draw a Polygon" function of the Frontal Editor allows the user to select a group of frontal line points within a user defined polygon for deletion. The user draws the polygon by clicking the left mouse button to choose the current mouse pointer position as a point in the polygon, clicking the middle mouse button to remove the previously selected point, and clicking the right mouse button to close up the polygon. The polygon must be drawn using at least three points. When the polygon has been drawn, all frontal line points within the polygon will cycle through the ten standard colors and the user will be prompted to keep or delete the selected points. Clicking the left or middle mouse buttons exits the function keeping the selected frontal line points. Clicking the right mouse button exits the function deleting the selected frontal line points.

2.3.7 Draw a Front Line

The "Draw a Front Line" function of the Frontal Editor allows the user to add new front lines to a displayed frontal image. If no frontal image is currently displayed, then one is created. Upon entering the function, the user is prompted for a front line thickness and then drawing mode is entered. When drawing the front line, clicking the left mouse button chooses the current mouse pointer position as an anchor point, clicking the middle mouse button removes the previously selected anchor point, and clicking the right mouse button exits the function saving any frontal lines drawn.

2.3.8 Frontal Editor Help

The "Frontal Editor Help" function of the Frontal Editor displays a short description of each of the functions available on the "Frontal Editor Menu". Clicking a mouse button, while the mouse pointer is in the help window, will advance the text or exit the function if all text has been displayed.

2.4 Gulf Stream Edit

The "Gulf Stream Edit" function allows the user to load, edit, and save Gulf Stream files. Selecting the "Gulf Stream Edit" function brings up the "Gulf Stream Editor Menu" shown in Figure 2.4-1.

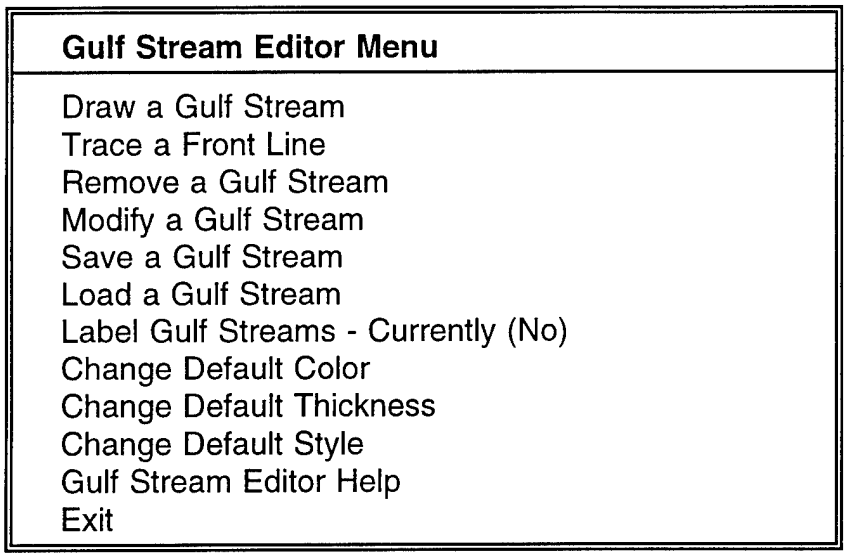


Figure 2.4-1 Gulf Stream Editor Menu

2.4.1 Draw a Gulf Stream

The "Draw a Gulf Stream" function of the Gulf Stream Editor allows the user to interactively draw a Gulf Stream line. Gulf Stream lines are drawn using the mouse pointer. Clicking the left mouse button selects the current mouse pointer position as a point in the Gulf Stream line. Clicking the middle mouse button removes the previously selected point. Clicking the right mouse button allows the user to change the drawing mode. The drawing mode choices are to toggle whether or not to connect points which are selected, continue drawing in the same mode, or to exit the function. Before starting to draw the line, the user is prompted for a graphic name to identify the line. This graphic name will also be used as the Gulf Stream file name. The Gulf Stream line is drawn in the current default color, thickness and line style.

2.4.2 Trace a Front Line

The "Trace a Front Line" function of the Gulf Stream Editor allows the user to draw a Gulf Stream line by automatically following an existing front line. The user is prompted to select the beginning and end points of the Gulf Stream line along a front line and to enter in a graphics name for the Gulf Stream line after the two points have been selected. Error messages are printed if no front lines are present or if the two points selected are not on the same front line. The Gulf Stream line is drawn in the current default color, thickness and line style.

2.4.3 Remove a Gulf Stream

The "Remove a Gulf Stream" function of the Gulf Stream Editor allows the user to remove a Gulf Stream line by name, remove all Gulf Stream lines at once, or exit the function. If a Gulf Stream line is removed by name, then that Gulf Stream line will cycle through the ten standard colors and the user will be prompted for confirmation.**2.4.4 Modify a Gulf Stream**

The "Modify a Gulf Stream" function of the Gulf Stream Editor allows the user to modify the attributes of, or trace and edit an existing Gulf Stream line. Selecting this function brings up a list of displayed Gulf Stream lines. After the desired Gulf Stream line is selected, the "Modification Mode" menu shown in Figure 2.4.4-1 is displayed.

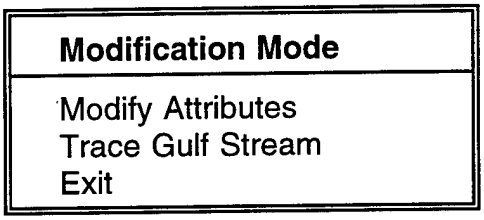


Figure 2.4.4-1 Modification Mode Menu

2.4.4.1 Modify Attributes

The "Modify Attributes" function of the "Modification Mode" menu allows the user to modify the color, thickness, and line style of the selected existing Gulf Stream line.

2.4.4.2 Trace Gulf Stream

The "Trace Gulf Stream" function of the "Modification Mode" menu allows the user to delete, add, disconnect, and redraw portions of the selected existing Gulf Stream line. When this function is selected, the "Trace Options" menu shown in Figure 2.4.4.2-1 is displayed.

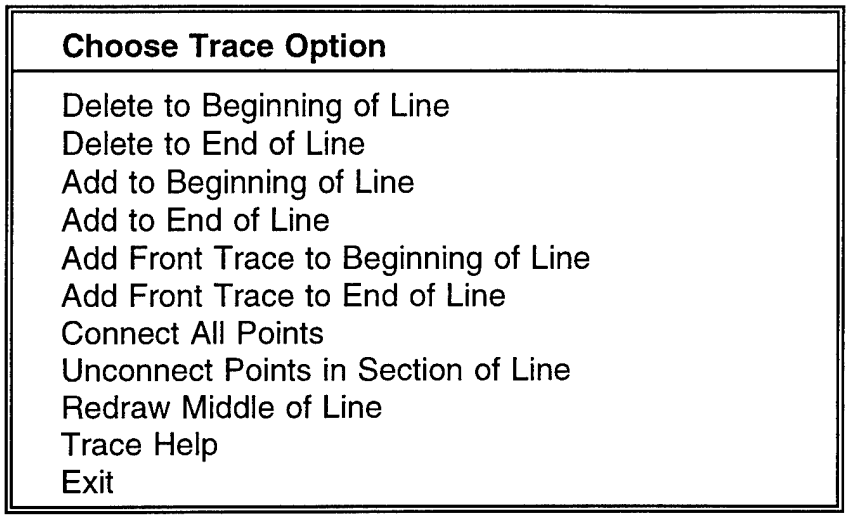


Figure 2.4.4.2-1 Trace Options Menu

2.4.4.2.1 Delete to Beginning of Line

The "Delete to Beginning of Line" function of the "Trace Options" menu allows the user to delete the portion of the selected Gulf Stream line before the selected mouse pointer position. The mouse pointer position is selected by clicking the left mouse button. If the right mouse button is clicked, the function is exited. After the portion of the line to be deleted has been selected, it is highlighted by cycling it through the ten standard colors and the user is prompted for confirmation. Deletion is confirmed by clicking the right mouse button. Deletion is canceled by clicking the left or middle mouse buttons.

2.4.4.2.2 Delete to End of Line

The "Delete to End of Line" function of the "Trace Options" menu allows the user to delete the portion of the selected Gulf Stream line following the selected mouse pointer position. The mouse pointer position is selected by clicking the left mouse button. If the right mouse button is clicked, the function is exited. After the portion of the line to be deleted has been selected, it is highlighted by cycling it through the ten standard colors and the user is prompted for confirmation. Deletion is confirmed by clicking the right mouse button. Deletion is canceled by clicking the left or middle mouse buttons.

2.4.4.2.3 Add to Beginning of Line

The "Add to Beginning of Line" function of the "Trace Options" menu allows the user to add to the beginning of the selected Gulf Stream line. When this function is selected, draw mode is entered with the original beginning point of the Gulf Stream

line as an undeletable starting point. New points are selected with the mouse pointer by clicking the left mouse button. The previously selected point can be removed by clicking the middle mouse button. The drawing mode can be changed by clicking the right mouse button. The drawing mode choices are to toggle whether or not to connect points which are selected, continue drawing in the same mode, or to exit the function.

2.4.4.2.4 Add to End of Line

The "Add to End of Line" function of the "Trace Options" menu allows the user to add to the end of the selected Gulf Stream line. When this function is selected, draw mode is entered with the original end point of the Gulf Stream line as an undeletable starting point. New points are selected with the mouse pointer by clicking the left mouse button. The previously selected point can be removed by clicking the middle mouse button. The drawing mode can be changed by clicking the right mouse button. The drawing mode choices are to toggle whether or not to connect points which are selected, continue drawing in the same mode, or to exit the function.

2.4.4.2.5 Add Front Trace to Beginning of Line

The "Add Front Trace to Beginning of Line" function of the "Trace Options" menu allows the user to trace a displayed front line and add that trace to the beginning of the selected Gulf Stream line. When this function is selected the user is instructed to select the first and last points in the front line to be traced and a trace increment. After the front line has been traced, the last point selected is connected to the beginning of the selected Gulf Stream line, and the first point selected becomes the new starting point for the Gulf Stream line. If the two points selected are not part of the same front line or if a front line is not present in the area where the either of the two points is selected, an error message is displayed and the function is exited without modifying the Gulf Stream line.

2.4.4.2.6 Add Front Trace to End of Line

The "Add Front Trace to End of Line" function of the "Trace Options" menu allows the user to trace a displayed front line and add that trace to the end of the selected Gulf Stream line. When this function is selected, the user is instructed to select the first and last points in the front line to be traced and a trace increment. After the front line has been traced, the first point selected is connected to the end of the selected Gulf Stream line and the last point selected becomes the new end point for the Gulf Stream line. If the two points selected are not part of the same front line or if a front line is not present in the area where the either of the two points is selected, an error message is displayed and the function is exited without modifying the Gulf Stream line.

2.4.4.2.7 Connect All Points

The "Connect All Points" function of the "Trace Options" menu allows the user to temporarily connect the unconnected points of the selected Gulf Stream line. The user then clicks a mouse button while the mouse pointer is on a continuation bar in the main menu to remove the temporary connection and exit the function.

2.4.4.2.8 Unconnect Points in Section of Line

The "Unconnect Points in Section of Line" function of the "Trace Options" menu allows the user to designate a section of the selected Gulf Stream line in which to switch all of the points to being unconnected. The two point are selected using the mouse pointer and clicking the left mouse button. If the right mouse button is clicked instead, the function is exited with no action taken.

2.4.4.2.9 Redraw Middle of Line

The "Redraw Middle of Line" function of the "Trace Options" menu allows the user to select a section of line to be redrawn. The user selects the beginning point of the section to be redrawn with the mouse pointer and clicking the left mouse button. The end point of the section is selected in the same manner. Clicking the right button while selecting either point will exit the function with no action being taken. If the user selected end point precedes the beginning point in the drawing sequence of the Gulf Stream, the beginning and end points are exchanged. After the beginning and end points have been selected, draw mode is entered with the beginning point being the undeletable starting point. New points are selected with the mouse pointer by clicking the left mouse button. The previously selected point can be removed by clicking the middle mouse button. The drawing mode can be changed by clicking the right mouse button. The drawing mode choices are to toggle whether or not to connect points which are selected, continue drawing in the same mode, or to exit the function. Exiting the function automatically connects the last selected point to the end point.

2.4.4.2.10 Trace Help

The "Trace Help" function of the "Trace Options" menu displays a short description of each of the functions available on the "Trace Options" menu. Clicking a mouse button, while the mouse pointer is in the help window, will advance the text or exit the function if all text has been displayed.

2.4.5 Save a Gulf Stream File

The "Save a Gulf Stream File" function of the Gulf Stream Editor allows the user to save the point information for a selected Gulf Stream line to a file in Gulf Stream format. The desired Gulf Stream line is selected by name from a list and the

information is written out to a file with a name of <graphic_name>.gs. If a file by that name already exists, the user is prompted whether or not to overwrite the file or to save the Gulf Stream line to a file with a new name. The Gulf Stream file format is as follows:

Header (Record 1)

I5	Number of positions contained in the file

Positions (Record 2 - EOF)

F10.4 Latitude		(+ North, - South)
F10.4 Longitude		(+ East, - West)
A2	Source Code	(I image, A altimeter, O other)

2.4.6 Load a Gulf Stream File

The "Load a Gulf Stream File" function of the Gulf Stream Editor allows the user to load the point information for a Gulf Stream line from a file in Gulf Stream format. The desired Gulf Stream file is selected by name from a list and the information is read into a gulf steam line with the same graphics name as the file name. The Gulf Stream file format is as follows:

Header (Record 1)

I5	Number of positions contained in the file

Positions (Record 2 - EOF)

F10.4 Latitude		(+ North, - South)
F10.4 Longitude		(+ East, - West)
A2	Source Code	(I image, A altimeter, O other)

2.4.7 Label Gulf Streams

The "Label Gulf Streams" function of the Gulf Stream Editor is a toggle menu selection used to determine whether or not to display the graphics names of displayed Gulf Streams. If the toggle is marked "Currently (Yes)" then the graphics names are displayed next to the starting points of the Gulf Stream lines. Having the labels displayed is useful when selecting a Gulf Stream line to be removed, modified, or saved to a file.

2.4.8 Change Default Color

The "Change Default Color" function of the Gulf Stream Editor allows the user to select the color in which new Gulf Stream lines will be drawn from among ten standard colors, see Appendix III.

2.4.9 Change Default Thickness

The "Change Default Thickness" function of the Gulf Stream Editor allows the user to select the thickness at which new Gulf Stream lines will be drawn. Possible thickness values are integer between one and twenty inclusive.

2.4.10 Change Default Style

The "Change Default Style" function of the Gulf Stream Editor allows the user to select the line style to be used in drawing new Gulf Stream lines from a standard set of line styles.

2.4.11 Gulf Stream Editor Help

The "Gulf Stream Editor Help" function of the Gulf Stream Editor displays a short description of each of the functions available on the "Gulf Stream Editor Menu". Clicking a mouse button, while the mouse pointer is in the help window, will advance the text or exit the function if all text has been displayed.

2.5 Eddy Edit

The "Eddy Edit" function allows the user to load, edit, and save eddy files. Selecting the "Eddy Edit" function brings up the "Eddy Editor Menu" shown in Figure 2.5-1.

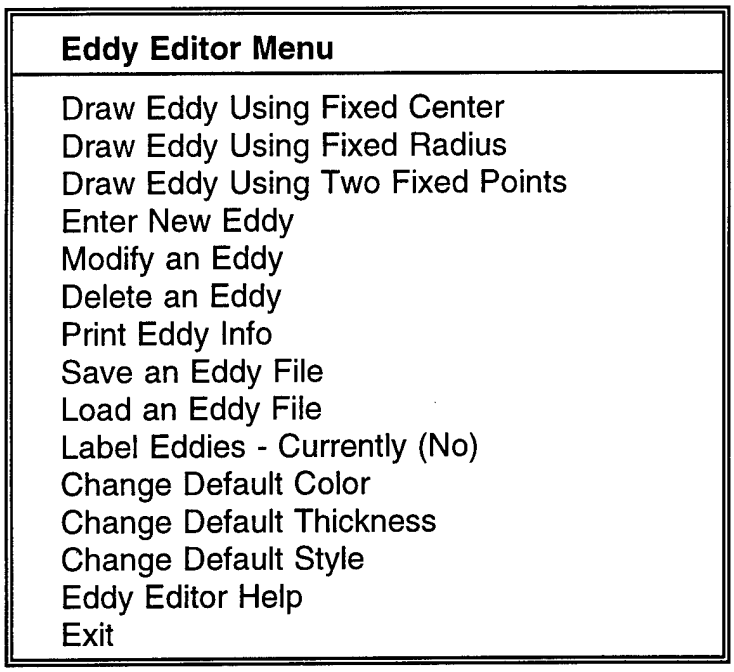


Figure 2.5-1 Eddy Editor Menu

2.5.1 Draw Eddy Using Fixed Center

The "Draw Eddy Using Fixed Center" function of the Eddy Editor allows the user to draw an eddy with a fixed center and a radius that varies with the position of the mouse pointer. When this function is entered, the user is prompted for whether a cold or warm eddy is being drawn, for a source code and a graphics name which automatically incorporates the eddy type and source code. The user then selects the fixed center point with the mouse pointer and clicking the left mouse button. The radius of the eddy then automatically shrinks and expands with the position of the mouse pointer. Clicking the left mouse button selects a new fixed center point. Clicking the middle mouse button exits the function saving the eddy which has been drawn. Clicking the right mouse button exits the function without saving the eddy.

2.5.2 Draw Eddy Using Fixed Radius

The "Draw Eddy Using Fixed Radius" function of the Eddy Editor allows the user to draw an eddy with a fixed radius and a center that varies with the position of the mouse pointer. When this function is entered, the user is prompted for whether a cold or warm eddy is being drawn, for the radius in kilometers, for a source code and a graphics name which automatically incorporates the eddy type and source code. The user then selects the center point with the mouse pointer. Clicking the middle mouse

button exits the function saving the eddy which has been drawn. Clicking the right mouse button exits the function without saving the eddy.

2.5.3 Draw Eddy Using Two Fixed Points

The "Draw Eddy Using Two Fixed Points" function of the Eddy Editor allows the user to draw an eddy based on two fixed and a center and radius that vary with the position of the mouse pointer. When this function is entered, the user is prompted for whether a cold or warm eddy is being drawn, for a source code and a graphics name which automatically incorporates the eddy type and source code. The user then selects the two fixed points with the mouse pointer and clicking the left mouse button. The center and radius of the eddy then automatically moves with the position of the mouse pointer being a third point on the curve. Clicking the middle mouse button exits the function saving the eddy which has been drawn. Clicking the right mouse button exits the function without saving the eddy.

2.5.4 Enter New Eddy

The "Enter New Eddy" function of the Eddy Editor allows the user to draw an eddy by entering its center latitude and longitude and its radius in kilometers. The user is also prompted for whether it is a cold or warm eddy, for a source code, and for a graphics name which automatically incorporates the eddy type and source.

2.5.5 Modify an Eddy

The "Modify an Eddy" function of the Eddy Editor allows the user to modify the color, thickness and line style of an eddy selected by name.

2.5.6 Delete an Eddy

The "Delete an Eddy" function of the Eddy Editor allows the user to remove a displayed eddy from the working image by name, delete all eddies from the working image, or exit the function. If an eddy is selected to delete by name, then that eddy cycles through the ten standard colors and the user is prompted for deletion confirmation.

2.5.7 Print Eddy Info

The "Print Eddy Info" function of the Eddy Editor allows the user to view the information on all displayed eddies. The information on all eddies is displayed to new window on the screen. The information that is displayed is the eddy graphics name, the position and radius in screen coordinates, the position and radius in geographic coordinates, the eddy type and the eddy source. The user is prompted to click a

mouse button while the mouse pointer is within a red rectangle in the main window to continue.

2.5.8 Save an Eddy File

The "Save an Eddy File" function of the Eddy Editor allows the user to save the information for all displayed eddies to a file in eddy format. The user is prompted for a root file name and the information is written out to a file with a name of <root_name>.eddy. If a file by that name already exists, the user is prompted whether or not to overwrite the file. The format of eddy files is as follows:

Header (Record 1)

I5	Number of eddies contained in the file

Positions (Record 2 - EOF)

F10.4	Center Latitude	(+ North, - South)
F10.4	Center Longitude	(+ East, - West)
F10.4	Radius	(Kilometers)
A2	Eddy Type	(W warm, C cold, U undefined)
A3	Source Code	(2 character code)

2.5.9 Load an Eddy File

The "Load an Eddy File" function of the Eddy Editor allows the user to load the information for several eddies from a file in eddy format. The desired eddy file is selected by name from a list and the information is read into new eddies with graphics names which contain the file name. The format of eddy files is as follows:

Header (Record 1)

I5	Number of eddies contained in the file

Positions (Record 2 - EOF)

F10.4	Center Latitude	(+ North, - South)
F10.4	Center Longitude	(+ East, - West)
F10.4	Radius	(Kilometers)
A2	Eddy Type	(W warm, C cold, U undefined)
A3	Source Code	(2 character code)

2.5.10 Label Eddies

The "Label Eddies" function of the Eddy Editor is a toggle for whether or not to display the graphics names of displayed eddies. If the toggle is marked "Currently (Yes)" then the graphics names are displayed next to the center points of the eddies. Having the labels displayed is useful when selecting an eddy to be removed or modified.

2.5.11 Change Default Color

The "Change Default Color" function of the Eddy Editor allows the user to select the color in which new eddies will be drawn from among ten standard colors.

2.5.12 Change Default Thickness

The "Change Default Thickness" function of the Eddy Editor allows the user to select the thickness at which new eddies will be drawn. Possible thickness values are integer numbers between one and twenty inclusive.

2.5.13 Change Default Style

The "Change Default Style" function of the Eddy Editor allows the user to select the line style to be used in drawing new eddies from a standard set of line styles.

2.5.14 Eddy Editor Help

The "Eddy Editor Help" function of the Eddy Editor displays a short description of each of the functions available on the "Eddy Editor Menu". Clicking a mouse button, while the mouse pointer is in the help window, will advance the text or exit the function if all text has been displayed.

2.6 Write a Picture File

The "Write a Picture File" function provides the user with the ability to save a modified image from the screen to a file under a user selected name. The picture file name must end in ".pic" so that the "Display Picture Files and Read One In" function under the File Menu can find it and read it in. Picture file are store in the image directory. If a file already exists in the image directory with the name entered, then the user is prompted whether or not to overwrite the file.

2.7 Increase Magnification

The "Increase Magnification" function allows the user to "zoom up" on a specific area of the working image for better resolution of features. A zoom indicator box is displayed within the image to mark the portion of the image to be magnified. The indicator box is moved with the mouse with the mouse pointer being at its center. Clicking the left mouse button activates the magnification. Clicking the right mouse button exits the function without magnifying the image. The magnified image replaces the existing image. The image can be magnified twice for a total magnification of four times the original image size. All functions work, and all graphics follow the image for all levels of magnification.

2.8 Decrease Magnification

The "Decrease Magnification" function allows the user to return to the image displayed before the "Increase Magnification" function was used. The previous image replaces the magnified image as the current working image. All functions work, and all graphics follow the image for all levels of magnification.

2.9 Annotate

The "Annotate" function allows the user to display text strings within an image. The text string, size, color, font, orientation and background color of the annotation are user selectable. Selecting the "Annotate" function brings up the "Annotate Menu" shown in Figure 2.9-1.

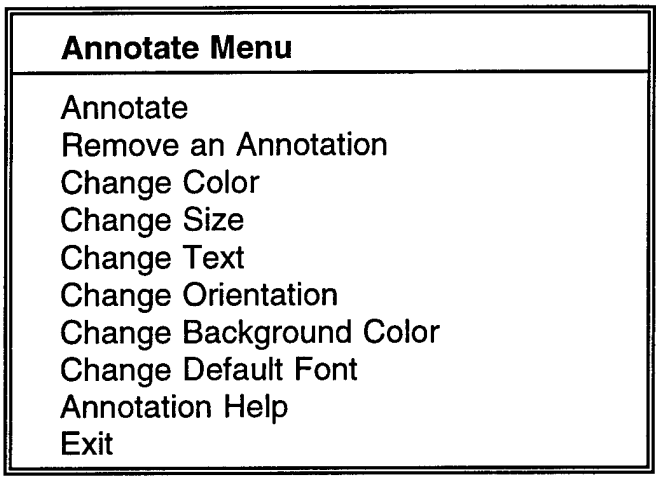


Figure 2.9-1 Annotation Menu

2.9.1 Annotate

The "Annotate" function of the "Annotate Menu" allows the user to select where in the image the text string will be written. The text string and instructions are displayed in the main window. The string is positioned by moving a box of the same size, color and orientation as the text string around on the screen using the mouse pointer. Clicking the left mouse button will deposit the string at the current box position. If the background color is other than clear, then the area where the box is shaded with the background color before the text string is written to the image.

2.9.2 Remove an Annotation

The "Remove an Annotation" function of the "Annotation Menu" allows the user to remove an annotation from the image by its graphics name. When the "Remove an

Annotation" function is selected, a list of all currently displayed annotations is displayed by their graphics names in the order in which they were displayed. The user can select an individual annotation to remove, remove all annotations at once, or exit the function.

2.9.3 Change Color

The "Change Color" function of the "Annotation Menu" allows the user to choose the color in which new annotations will be written from a list of ten standard colors, see Appendix III.

2.9.4 Change Size

The "Change Size" function of the "Annotation Menu" allows the user to choose the size in which new annotations will be written. Available sizes are integers between one and twenty inclusive.

2.9.5 Change Text

The "Change Text" function of the "Annotation Menu" allows the user to enter in the string to be annotated. The string is prompted for and typed into the main window.

2.9.6 Change Orientation

The "Change Orientation" function of the "Annotation Menu" allows the user to choose the angle at which new annotations will be displayed. Available angles are integer degrees between 0 and 345 inclusive at 15 degree increments.

2.9.7 Change Background Color

The "Change Background Color" function of the "Annotation Menu" allows the user to select the color of the box behind any new annotations from a list of ten standard colors or a clear background. Selecting the clear background will cause the annotation to be displayed without a background box.

2.9.8 Change Default Font

The "Change Default Font" function of the "Annotation Menu" allows the user to select the font in which new annotations will be displayed from a list of seventeen standard fonts.

2.9.9 Annotation Help

The "Annotation Help" function of the "Annotation Menu" displays a short description of each of the functions available on the "Annotation Menu". Clicking a mouse button, while the mouse pointer is in the help window, will advance the text or exit the function if all text has been displayed.

2.10 Save Graphics

The "Save Graphics" function allows the user to save Gulf Stream lines, eddies, and annotations to files in standard graphics editor format. These files can be read back in from either the "Load Graphics" function in Section 2.11 or from a similar function in the Graphics Editor under the Display Menu of NSIPS.

2.11 Load Graphics

The "Load Graphics" function allows the user to load Gulf Stream lines, eddies, and annotations from files in standard graphics editor format. These file can be saved from either the "Save Graphics" function in Section 2.10 or from a similar function in the Graphics Editor under the Display Menu of NSIPS.

2.12 Change Image Directory

The "Change Image Directory" function allows the user to select a new directory for saving and loading file. The directory name must end in a "/".

2.13 Reset SAMAS

The "Reset SAMAS" function allows the user to change the working image and clear out all working variables for SAMAS. The user is prompted for confirmation before the function is executed.

2.14 SAMAS Help

The "SAMAS Help" function displays a short description of each of the functions available on the SAMAS main menu. Clicking a mouse button, while the mouse pointer is in the help window, will advance the text or exit the function if all text has been displayed.

3.0 ERROR CONDITIONS AND DIAGNOSTICS

All error messages produced by the SAMAS software are displayed in either the MAIN window or in the shell tool from which NSIPS was started. All of the error messages

are self descriptive. For error messages related to files, the user should make sure that they have read and write access to both the file and the directory in which it is contained and that there is sufficient disk space available for the file.

4.0 GLOSSARY

NRL	Naval Research Laboratory
NSIPS	NRL Satellite Image Processing System
PV-WAVE	Precision Visual - Workstation Analysis and Visualization Environment
SAMAS	Semi-Automated Mesoscale Analysis System

Appendix B

NSIPS Image File Header Format

Bytes	Item	Type	Description
0-3	Version number	I 4	Currently at 1
4-7	Data Type	I 4	1 - Integer 1 (byte) 2 - Integer 2 3 - Integer 4 4 - Float 4
8-11	Samples	I 4	
12-15	Lines	I 4	
16-19	Bands	I 4	Not currently used
20-23	Times	I 4	Not currently used
24-27	Projection	I 4	1 - Rectilinear 2 - Mercator 3 - Stereographic 4 - Polar Stereo
28-31	Grid size - x	I 4	
32-35	Grid size - y	I 4	
36-39	Ellipsoid	I 4	
40-43	Zone	I 4	
44-47	Minimum Value	I 4	For User defined
48-51	Maximum Value	I 4	scaling

F 4 data stored as two I 4

52-55	Latitude Int	I 4	North or Center
56-59	Latitude Division	I 4	
60-63	Longitude Int	I 4	West or Center
64-67	Longitude Division	I 4	
68-71	Deg/Pixel - Lat Int	I 4	
72-75	Deg/Pixel - Lat Div	I 4	
76-79	Deg/Pixel - Lon Int	I 4	
80-83	Deg/Pixel - Lon Div	I 4	
84-87	True Latitude Int	I 4	
88-91	True Latitude Div	I 4	
92-95	True Longitude Int	I 4	
96-99	True Longitude Div	I 4	
100-103	X Scale Int	I 4	
104-107	X Scale Divisor	I 4	
108-111	Y Scale Int	I 4	
112-115	Y Scale Divisor	I 4	

F 8 data stored as two I 4

116-119	Base Latitude Int	I 4	
120-123	Base Latitude Div	I 4	
124-127	Base Longitude Int	I 4	
128-131	Base Longitude Int	I 4	
132-135	False Easting Int	I 4	
136-139	False Easting Div	I 4	
140-143	False Northing Int	I 4	
144-147	False Northing Div	I 4	
148-187	Misc Int (10)	I 4	
188-227	Misc Divisor (10)	I 4	
228-231	Minimum Map X Int	I 4	
232-235	Minimum Map X Div	I 4	
236-239	Maximum Map X Int	I 4	
240-243	Maximum Map X Div	I 4	
244-247	Minimum Map Y Int	I 4	
248-251	Minimum Map Y Div	I 4	
252-255	Maximum Map Y Int	I 4	
256-259	Maximum Map Y Div	I 4	
260-263	Minimum Map Lat Int	I 4	
264-267	Minimum Map Lat Div	I 4	
268-271	Maximum Map Long Int	I 4	
272-275	Maximum Map Long Div	I 4	
276-279	Minimum Map Lat Int	I 4	
280-283	Minimum Map Lat Div	I 4	
284-287	Maximum Map Long Int	I 4	
288-291	Maximum Map Long Div	I 4	
292-316	Comment 1	String	Not currently used
317-341	Comment 2	String	Not currently used
342-511	Spare		Not currently used

"F 4" and "f 8" data stored as two "I 4" can be converted back by dividing the variable labeled "Int" by the variable labeled "Div" or "Divisor". If the "Div" or "Divisor" is zero then the actual value is assumed to be zero.

Appendix C

.defines File

```
setenv SRCF /disks/sips3d2/run/imagepro/Source/F77
setenv SRCC /disks/sips3d2/run/imagepro/Source/C
setenv SRCI /disks/sips3d2/run/imagepro/Source/C_Include
setenv SRCU  /disks/sips3d2/sarah/source
setenv SAMASEXE /disks/sips3d2/sarah/exe
setenv SAMASDAT /disks/sips3d2/sarah/dat
setenv EXPERTEXE /disks/sips3d2/sarah/deliver
setenv EXPERTDAT /disks/sips3d2/sarah/deliver/dat
setenv EXPERTDAT2 /disks/sips3d2/sarah/deliver/dat
setenv EXPERTSRC /disks/sips3d2/sarah/deliver
```

Appendix D

Functions Listed with Required Executable Files, Source Files, and Data Files

SAMAS Menu
$SAMASEXE/SAMAS $SRCU/samas.pro

$SRCU/samas_command.lis

Dump Image Values
$SAMASEXE/dump_image
$SRCU/image_io.f
$SRCC/io_suntran.c
$SRCU/dump_image.c

Subsection/subsample Image
$SAMASEXE/ss_image
$SRCU/mapf.f
$SRCU/image_io.f
$SRCU/rcio.f
$SRCU/mapc.c
$SRCU/input.c
$SRCU/ss_image.c

Enter List - Eddy
$SAMASEXE/eddy_enterlist
$SRCU/mapf.f
$SRCU/mapc.c
$SRCU/eddy_enterlist.f

Enter List - Gulf Stream
$SAMASEXE/gs_enterlist
$SRCU/mapf.f
$SRCU/mapc.c
$SRCU/enterlist.lnk

Read OOC, Output NW and eddy list files
$SAMASEXE/reformat_ooc
$SRCU/reformat_ooc.c
$SRCU/input.c

Run Hough - Altimeter Gulf Stream Crossing
$SAMASEXE/houghgs
$SRCU/houghgs.c
$SRCU/input.c

$SAMASEXE/hough_gs
$SRCU/hough_gs.f

$SAMASDAT/gsmean.dat

Run Expert System

$EXPERTEXE/expsys

$EXPERTSRC/expert.c

$SRCU/input.c

$EXPERTEXE/wate

$EXPERTSRC/analysis.c

$EXPERTSRC/bload.c

$EXPERTSRC/bsave.c

$EXPERTSRC/build.c

$EXPERTSRC/classcom.c

$EXPERTSRC/classfun.c

$EXPERTSRC/commline.c

$EXPERTSRC/conscomp.c

$EXPERTSRC/constrct.c

$EXPERTSRC/deffacts.c

$EXPERTSRC/deffnctn.c

$EXPERTSRC/defglobl.c

$EXPERTSRC/defins.c

$EXPERTSRC/defrule.c

$EXPERTSRC/deftmcom.c

$EXPERTSRC/deftmfun.c

$EXPERTSRC/deftmlhs.c

$EXPERTSRC/deftmpsr.c

$EXPERTSRC/drive.c

$EXPERTSRC/drulebin.c

$EXPERTSRC/edbasic.c

$EXPERTSRC/edmain.c

$EXPERTSRC/edmisc.c

$EXPERTSRC/edstruct.c

$EXPERTSRC/edterm.c

$EXPERTSRC/engine.c

$EXPERTSRC/evaluatn.c

$EXPERTSRC/expressn.c

$EXPERTSRC/factcom.c

$EXPERTSRC/factmngr.c

$EXPERTSRC/generate.c

$EXPERTSRC/genrcbin.c

$EXPERTSRC/genrccmp.c

$EXPERTSRC/genrccom.c

$EXPERTSRC/genrcfun.c

$EXPERTSRC/inscom.c

$EXPERTSRC/insfun.c
$EXPERTSRC/insquery.c
$EXPERTSRC/intrfile.c
$EXPERTSRC/lgcldpnd.c
$EXPERTSRC/match.c
$EXPERTSRC/math.c
$EXPERTSRC/memory.c
$EXPERTSRC/msgcom.c
$EXPERTSRC/msgfun.c
$EXPERTSRC/multivar.c
$EXPERTSRC/objbin.c
$EXPERTSRC/objcmp.c
$EXPERTSRC/parsutil.
$EXPERTSRC/reorder.c
$EXPERTSRC/reteutil.c
$EXPERTSRC/retract.c
$EXPERTSRC/router.c
$EXPERTSRC/rulecom.c
$EXPERTSRC/ruleprsr.c
$EXPERTSRC/scanner.c
$EXPERTSRC/spclform.c
$EXPERTSRC/strings.c
$EXPERTSRC/symbol.c
$EXPERTSRC/sysdep.c
$EXPERTSRC/sysio.c
$EXPERTSRC/syspred.c
$EXPERTSRC/sysprime.c
$EXPERTSRC/syssecnd.c
$EXPERTSRC/textpro.c
$EXPERTSRC/utility.c
$EXPERTSRC/variable.c
$EXPERTSRC/eddies.c
$EXPERTSRC/initialize.c
$EXPERTSRC/wate.c
$EXPERTSRC/readinput.c
$EXPERTSRC/mathe.c
$EXPERTSRC/regions.c
$EXPERTSRC/mapping.c
$EXPERTSRC/modes.c
$EXPERTSRC/nrmlz.c
$EXPERTSRC/explan.c
$EXPERTSRC/returns.c
$EXPERTSRC/gs_fc.c
$EXPERTSRC/final-output.c

$EXPERTSRC/wate-interface-stubs.c
$EXPERTSRC/wate-stubs.c
$EXPERTSRC/mapf.f
$EXPERTSRC/mapc.c

$EXPERTSRC/analysis.h
$EXPERTSRC/bload.h
$EXPERTSRC/bsave.h
$EXPERTSRC/build.h
$EXPERTSRC/ceofdef.h
$EXPERTSRC/classcom.h
$EXPERTSRC/classfun.h
$EXPERTSRC/clips.h
$EXPERTSRC/clipsmem.h
$EXPERTSRC/cmptblty.h
$EXPERTSRC/colors.h
$EXPERTSRC/commline.h
$EXPERTSRC/conscomp.h
$EXPERTSRC/constant.h
$EXPERTSRC/constrct.h
$EXPERTSRC/default.h
$EXPERTSRC/deffacts.h
$EXPERTSRC/deffnctn.h
$EXPERTSRC/defglobl.h
$EXPERTSRC/defins.h
$EXPERTSRC/defrule.h
$EXPERTSRC/deftempl.h
$EXPERTSRC/deftmfun.h
$EXPERTSRC/deftmlhs.h
$EXPERTSRC/deftmpsr.h
$EXPERTSRC/drive.h
$EXPERTSRC/ed.h
$EXPERTSRC/engine.h
$EXPERTSRC/evaluatn.h
$EXPERTSRC/explan.h
$EXPERTSRC/expressn.h
$EXPERTSRC/extobj.h
$EXPERTSRC/facts.h
$EXPERTSRC/generate.h
$EXPERTSRC/genrcbin.h
$EXPERTSRC/genrccmp.h
$EXPERTSRC/genrccom.h
$EXPERTSRC/genrcfun.h
$EXPERTSRC/globals.h

$EXPERTSRC/inscom.h
$EXPERTSRC/insfun.h
$EXPERTSRC/insquery.h
$EXPERTSRC/intrfile.h
$EXPERTSRC/lgcldpnd.h
$EXPERTSRC/match.h
$EXPERTSRC/msgcom.h
$EXPERTSRC/msgfun.h
$EXPERTSRC/multivar.h
$EXPERTSRC/network.h
$EXPERTSRC/objbin.h
$EXPERTSRC/objcmp.h
$EXPERTSRC/object.h
$EXPERTSRC/parsutil.h
$EXPERTSRC/reorder.h
$EXPERTSRC/reteutil.h
$EXPERTSRC/retract.h
$EXPERTSRC/router.h
$EXPERTSRC/ruleprsr.h
$EXPERTSRC/scanner.h
$EXPERTSRC/setup.h
$EXPERTSRC/shrtlnkn.h
$EXPERTSRC/spclform.h
$EXPERTSRC/symbol.h
$EXPERTSRC/sysdep.h
$EXPERTSRC/sysprime.h
$EXPERTSRC/utility.h
$EXPERTSRC/variable.h
$EXPERTSRC/wate-interface.h

$EXPERTDAT/nomgs.lat
$EXPERTDAT/nomgs.long
$EXPERTDAT/parms.dat
$EXPERTDAT/regions.dat
$EXPERTDAT/w_map_grid.img
$EXPERTDAT/expert.pro
$EXPERTDAT/help.me
$EXPERTDAT2/gsmean.int
$EXPERTDAT2/ev1.eig
$EXPERTDAT2/ev2.eig
$EXPERTDAT2/ev3.eig
$EXPERTDAT2/ev4.eig
$EXPERTDAT2/ev5.eig
$EXPERTDAT2/ev6.eig

$EXPERTDAT2/ev7.eig
$EXPERTDAT2/ev8.eig
$EXPERTDAT2/ev9.eig
$EXPERTDAT2/ev10.eig
$EXPERTDAT2/ev11.eig
$EXPERTDAT2/ev12.eig
$EXPERTDAT2/ev13.eig
$EXPERTDAT2/ev14.eig
$EXPERTDAT2/ev15.eig
$EXPERTDAT2/ev16.eig
$EXPERTDAT2/ev17.eig
$EXPERTDAT2/ev18.eig
$EXPERTDAT2/ev19.eig
$EXPERTDAT2/ev20.eig
$EXPERTDAT/k-stru.clp
$EXPERTDAT/wcrrules.clp
$EXPERTDAT/ccrrules.clp
$EXPERTDAT/explain.clp

Create Edge Image (Cluster Shade)
$SAMASEXE/edge
$SRCU/edge.c
$SRCU/input.c

$SAMASEXE/remove
$SRCU/remove.c

$SAMASEXE/cluster_shade
$SRCU/image_io.f
$SRCU/cluster_shade.f
$SRCC/io_suntran.c

$SAMASEXE/cluster_test
$SRCU/image_io.f
$SRCU/cluster_test.f
$SRCC/io_suntran.c

$SAMASEXE/dilate_clean
$SRCU/image_io.f
$SRCU/dilate_clean.f
$SRCC/io_suntran.c

$SAMASEXE/thin_line
$SRCU/image_io.f

```
                $SRCU/thin_line.f
                $SRCC/io_suntran.c

Label Edge Image (Relaxation method)
        $SAMASEXE/label_driver
                $SRCU/label_driver.c
                $SRCU/input.c

                $SAMASEXE/remove
                        $SRCU/remove.c

                $SAMASEXE/edge_extract
                        $SRCU/image_io.f
                        $SRCU/edge_extract.c
                        $SRCC/io_suntran.c

                $SAMASEXE/a1
                        $SRCU/a1.c
                        $SRCU/image_io.f
                        $SRCC/io_suntran.c

                        $SAMASDAT/Angle.Dat
                        $SAMASDAT/Dist.Dat

                $SAMASEXE/a2
                        $SRCU/a2.c
                        $SRCU/image_io.f
                        $SRCC/io_suntran.c

                        $SAMASDAT/Angle.Dat
                        $SAMASDAT/Dist.Dat

                $SAMASEXE/rdata
                        $SRCU/rdata.c
                        $SRCU/image_io.f
                        $SRCC/io_suntran.c

                $SAMASEXE/label
                        $SRCU/label.c
                        $SRCU/image_io.f
                        $SRCC/io_suntran.c

                $SAMASEXE/splitlabeljoin
                        $SRCU/splitlabeljoin.c
```

```
                $SRCU/mapf.f
                $SRCU/mapc.c

Label Edge Image (Topographic method)
        $SAMASEXE/label_driver2
                $SRCU/label_driver2.c
                $SRCU/input.c

                $SAMASEXE/remove
                        $SRCU/remove.c

                $SAMASEXE/rdata_new
                        $SRCU/mapf.f
                        $SRCU/mapc.c
                        $SRCU/rdata_new.c

                $SAMASEXE/demo4
                        $SRCU/demo4.c


Process Regions
        $SAMASEXE/rs_driver
                $SRCU/rs_driver.c
                $SRCU/input.c

                $SAMASEXE/rs5
                        $SRCC/io_suntran.c
                        $SRCU/rs5.f
                        $SRCU/image_io.f

                $SAMASEXE/rs1024
                        $SRCU/rs1024.f
                        $SRCU/image_io.f
                        $SRCC/io_suntran.c

Run CEOF - Gulf Stream Formation
        $SAMASEXE/ceof
                $SRCU/ceof.c
                $SRCU/input.c

                $SAMASEXE/remove
                        $SRCU/remove.c
                $SAMASEXE/gs_extract
                        $SRCU/mapf.f
```

```
                $SRCU/mapc.c
                $SRCU/gs_extract

        $SAMASEXE/interp
                $SRCU/interp.f

                $SAMASDAT/eigvect.dat
                $SAMASDAT/primary.dat
                $SAMASDAT/second.dat

        $SAMASEXE/prepopto
                $SRCU/prepopto.f

                $SAMASDAT/primary.dat
                $SAMASDAT/second.dat

        $SAMASEXE/opto
                $SRCU/opto.f

                $SAMASDAT/eigvect.dat

Run Hough - Eddy Detect
        $SAMASEXE/hough
                $SRCU/hough.c
                $SRCU/input.c

        $SAMASEXE/remove
                $SRCU/remove.c

        $SAMASEXE/edge_extract
                $SRCU/edge_extract.c
                $SRCU/image_io.f
                $SRCC/io_suntran.c

        $SAMASEXE/eddydetect
                $SRCU/mapf.f
                $SRCU/mapc.c
                $SRCU/eddydetect.f

Apply Image Mask
        $SAMASEXE/mask
                $SRCU/image_io.f
                $SRCU/mask.c
                $SRCU/input.c
```

$SRCC/io_suntran.c

Convert Gulf Stream for Connect
$SAMASEXE/gs_code_I
$SRCU/gs_code_I.f

System Command
$SAMASEXE/command
$SRCU/command.c
$SRCU/input.c

notes:

Some executables are driver routines which require other executables, as well as source and/or data files.

mapc.c required files
$SRCC/io_suntran.c

For io_suntran.c include files are contained in $SRCI and the executable $SAMASEXE/mk_rc_to_img is required. The file mk_rc_to_img is copied from /disks/sips3d2/run/imagepro/Bin/Sun/mk_rc_to_img.

mapf.f required files
$SRCF/xy_00_ll.f
$SRCF/xy_01_ll.f
$SRCFxy_02_ll.f
$SRCF/xy_03_ll.f
$SRCF/xy_04_ll.f
$SRCF/xy_05_ll.f
$SRCF/xy_06_ll.f
$SRCF/xy_tm_ll.f
$SRCF/xy_to_ll.f
$SRCF/ll_00_xy.f
$SRCF/ll_01_xy.f
$SRCF/ll_02_xy.f
$SRCF/ll_03_xy.f
$SRCF/ll_04_xy.f
$SRCF/ll_05_xy.f
$SRCF/ll_06_xy.f
$SRCF/ll_tm_xy.f
$SRCF/ll_to_xy.f

$SRCF/coord_conv.f
$SRCF/mapfuncts.f
$SRCF/fourier.f
$SRCF/ind_fourier.f
$SRCC/image_io.f
$SRCC/map_xy_ll.f

To compile the system use the following script and make files.

$SAMASEXE/exe.com (script file for other SAMAS modules)
- $SRCU/ceof.lnk
- $SRCU/enterlist.lnk
- $SRCU/dump_image.lnk
- $SRCU/edge.lnk
- $SRCU/hough.lnk
- $SRCU/mask.lnk
- $SRCU/hough_gs.lnk
- $SRCU/label.lnk
- $SRCU/label_new.lnk
- $SRCU/ss_image.lnk
- $SRCU/rs5.lnk

$EXPERTSRC/Make.wate (make file for expert system)